UNIVERZITA PARDUBICE
FAKULTA ELEKTROTECHNIKY
A INFORMATIKY

BAKALÁŘSKÁ PRÁCE

2010 Petr Hybler

UNIVERZITA PARDUBICE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

ZABEZPEČENÍ A SPRÁVA MENŠÍ FIREMNÍ SÍTĚ

BAKALÁŘSKÁ PRÁCE

AUTOR: Petr Hybler
VEDOUCÍ PRÁCE: Mgr. Tomáš Hudec

2010

UNIVERSITY OF PARDUBICE
FACULTY OF ELECTRICAL ENGINEERING
AND INFORMATICS

SECURING AND ADMINISTRATION OF SMALL COMPANY NETWORK

BACHELOR WORK

AUTHOR: Petr Hybler
SUPERVISOR: Mgr. Tomáš Hudec

2010

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2009/2010

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Petr HYBLER**
Osobní číslo:	**I07895**
Studijní program:	**B2646 Informační technologie**
Studijní obor:	**Informační technologie**
Název tématu:	**Zabezpečení a správa menší firemní sítě**
Zadávající katedra:	**Katedra informačních technologií**

Zásady pro vypracování:

Teoretická část:
Diskutujte zabezpečení menší firemní sítě, která obsahuje uživatelské stanice na platformě Windows, servery na platformě Linux a aktivní prvky CISCO.
Věnujte se rizikům napadení sítě zvenčí i zevnitř (od jejích uživatelů).
Praktická část:
Navrhněte a nainstalujte služby vhodné pro danou malou firmu na serveru dané sítě.
Věnujte se nastavení zabezpečení vybraných služeb podle závěrů teoretické části.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

- OSTÁLEK, Libor; KABELOVÁ, Alena. Velký průvodce protokoly TCP/IP a systémem DNS. 2. aktualiz. vyd. Praha: Computer Press, 2000. 426 s. ISBN 978-80-251-2236-5.

- HARRIS, Shon, et al. Hacking ? manuál hackera. 1. vyd. Praha: Grada Publishing, 2008. 400 s.

- SCAMBREY, Joel; McCLURE, Stuart; KURTZ, George. Hacking bez tajemství. 2. aktualiz. vyd. Praha: Computer Press, 2002. 625 s.

- VESELSKÝ, Jiří, et al. Linux ? Dokumentační projekt [online]. 3. aktualiz. vyd. Brno: Computer Press, 2003. 1020 s. Dostupný z WWW: <http://knihy.cpress.cz/DataFiles/Book/00000675/Download/K0819.pdf>.

- VRANÝ, Boleslav. Bezpečnost v digitálním věku [online]. 2004 [cit. 2009-10-21], s. 2-36. Dostupný z WWW: <http://www.bolekvrany.cz/downloads/security_cz.pdf>.

Vedoucí bakalářské práce: **Mgr. Tomáš Hudec**
Katedra informačních technologií

Datum zadání bakalářské práce: **15. ledna 2010**

Termín odevzdání bakalářské práce: **14. května 2010**

L.S.

prof. Ing. Simeon Karamazov, Dr.
děkan

Ing. Lukáš Regan, Ph.D.
vedoucí katedry

V Pardubicích dne 31. března 2010

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně Univerzity Pardubice.

V Pardubicích dne 7.12. 2009

Petr Hybler

Poděkování

Rád bych poděkoval vedoucímu bakalářské práce, Mgr. Tomáši Hudcovi, za rady, připomínky a návrhy týkající se bakalářské práce a za odborné vedení při výběru software a ostatních bezpečnostních prvků.

Abstrakt

Tato práce je zaměřena na zabezpečení menší firemní sítě, obsahující uživatelské stanice na platformě Windows, servery na platformě Linux a aktivní prvky CISCO. Práce se bude věnovat rizikům napadení zevnitř i zvenčí. Praktická část se věnuje navrhnutí a nainstalování služeb, vhodných pro danou malou firmu na serveru dané sítě a nastavení zabezpečení vybraných služeb podle závěrů teoretické části.

Klíčová slova

TCP/IP; Linux; bezpečnost; iptables; firewall, hacking;

Obsah

Úvod

Bakalářská práce je zaměřena na zabezpečení a správu menší firemní sítě. Cílem této bakalářské práce je diskutovat zabezpečení sítě, možných napadení zevnitř a zvenčí, od zaměstnanců firmy a prakticky nastínit nastavení sítě, instalaci služeb a především konfiguraci serveru a nainstalovaných služeb.

Teoretická část práce se zaměří především na obecný pohled bezpečnosti sítě, potenciální rizika a jejich případné řešení. V první části jsou popsány možné útoky na firemní síť. Druhá část diskutuje služby, které je vhodné a které naopak není vhodné používat a proč. V třetí části se zaměřuje na možné útoky z vnitřní strany sítě - ze strany běžného uživatele. Čtvrtá část se orientuje na napadení a útoky vedené zvenčí. V závěru teoretické části jsou shrnuta veškerá rizika, prevence před napadením a případná náprava průniku.

V úvodu praktické části je uvedena struktura sítě, instalace serveru a jsou zde uvedeny služby, které byly vybrány pro konkrétní instalaci a konfigurování. Druhá část je zaměřena na konkrétní konfigurace vybraných služeb s maximálním ohledem na bezpečnost. Třetí část se věnuje detailnímu zabezpečení celého komplexu služeb. V závěru jsou shrnuty přínosy a výsledky bakalářské práce a jsou zde uvedeny služby a jejich konfigurace zaměřující se především na bezpečnost. Vyhodnocení se zakládá na provedení konfigurace služeb a serveru odvíjející se od teoretických závěrů.

1 Bezpečnost

Bezpečnost firemní sítě je velmi důležitý prvek, jelikož dnešní doba je založena na virtuálních systémech. Bez zabezpečené sítě by firma mohla přijít nejen o zisky ale i o reputaci a potencionální zákazníky. Z tohoto důvodu je dobré brát bezpečnost velmi vážně a nepodceňovat potencionální rizika.

Zabezpečení je komplexní úkon, jenž vyžaduje spoustu úsilí a neustálý dohled. Existuje mnoho možností jak provést neoprávněný vstup do systému zevnitř sítě i zvenčí. Úroveň bezpečnosti sítě je tak vysoká, jak vysoké znalosti má administrátor, který má síť na starosti.

Útoky na síť mohou být dvojího druhu - vnitřní a vnější. Vnitřní útoky pocházejí od firemních zaměstnanců a často zde hrají největší roli špatně nastavená práva. Ovšem velké procento útoků je vedeno zvenčí. Nejprve je nutné si objasnit, čeho chce útočník dosáhnout a jaké formy "útoku" může použít. Primárním cílem crackera je získání superuživatelského přístupu do systému, což znamená získat heslo k účtu root. Tím, že získá tento účet, získává neomezené možnosti a má právo dělat v systému co se mu zlíbí.

1.1 Ovlivňování, manipulace a jiné psychologické hry crackerů

Snad nejrozšířenějším způsobem jak z někoho vylákat heslo je sociální inženýrství. Sociální inženýrství je druh psychologie aplikovaného na zaměstnance firmy. Cílem této techniky je vylákat z dotyčné osoby heslo s vytvořením určitého nátlaku na psychiku nebo city „oběti“. To může cracker zkusit několika cestami. Nejběžnější cestou je falšování identity emailem nebo přes telefon ale není nezvyklé, že i osobně se může cracker vydávat za někoho jiného. Do sociálního inženýrství spadá i například nátlak v podobě osobního zaujetí v dané věci, kdy cracker předstírá, že ho velmi zajímá „tělo“ programu a rád by se o něm dozvěděl co nejvíc. Tak může najít ve zdrojovém kódu bezpečnostní díry a později je využít k průniku do systému.

1.2 Programová manipulace

Asi druhým nejrozšířenějším způsobem, jak se dostat k autentizačním údajům a tím získat přístup do sítě, jsou trojské koně. Trojský kůň je část kódu, která upravuje funkci a běh programu tak, aby falšovala výstupy nebo odesílala určitá data crackerovi.

1.3 Fyzický útok

Většina administrátorů a jiných bezpečnostních techniků ve svých kancelářích opomínají důležitost bezpečnosti. Často vyhazují neskartované citlivé údaje, mají nalepené papírky s hesly, nezamykají své skříňky natož pak pracovní plochy. Tímto chováním umožňují útočníkům zjistit, jak se dostat do systému či jinak zneužít tyto údaje.

Pokud se útočník dostane do kanceláře a je tam sám, má téměř neomezené možnosti. Nejen, že si může vzít z koše veškeré dokumenty na pozdější prohlédnutí, ale také si může odnést dokumenty z nezamčené skříňky, či jen tak položené na pracovní ploše. Mezi dokumenty, po kterých se bude útočník poohlížet, patří zejména telefonní seznamy, bezpečnostní předpisy, interní firemní manuály a v neposlední řadě například seznam dovolených. Tyto dokumenty může útočník zneužít k technikám sociálního inženýrství nebo je využít při plánování nejvhodnější doby útoku. Obzvláště otrlý útočník se pokusí ukrást pevné disky nebo celé počítače.

Administrátor by měl dbát na zabezpečení svého terminálu na monitoru. Pokud se útočník dostane k jeho počítači bez dozoru, nic nevidí raději, než nezamčený monitor. Aby k těmto situacím nedocházelo, je třeba chránit svůj počítač zvolením dobrého spořiče. Ten by měl být takový, aby po zamčení nebylo nic vidět a nešlo zamčení nijak obejít.

Proti fyzickému útoku je třeba zavést další opatření. I když bude mít administrátor všechny dokumenty bezpečně uloženy, zamčený monitor se silným heslem pro při přihlášení, může se útočník dostat do počítače. Určitě s sebou bude mít disketu, CD-ROM nebo USB flash disk. Z těchto zařízení je možné zavést vlastní operační systém. Jako příklad takovéto linuxové distribuce, kterou lze nabootovat z pár disket, je Trinux. Tato distribuce v sobě obsahuje několik síťových skenerů, analyzátor síťových paketů a jiné bezpečnostní nástroje.

Pokud nebude v BIOSu daného počítače zakázáno bootování z jiných, než přesně stanovených diskových jednotek, má útočník téměř vyhráno. V BIOSu by měly být úplně vyma-

zány bootovatelné položky jako CD-ROM, disketa a USB disky. Ani toto by nezabránilo útočníkovi znovu změnit bootování na požadované médium. Nejlepší ochranou je zakázání bootování z jiných než stanovených medií a zaheslování BIOSu. Toto heslo může mít maximálně 7 znaků, jde obejít hardwarovým resetováním BIOSu, ale alespoň tím velmi zkomplikujete útočníkovu snahu.

1.4 Napadení přes síť

Většina útoků vedených na firemní síť pochází zvenčí. Existuje mnoho možností, jak vzdáleně napadnout firemní síť. V linuxových ale i jiných operačních systémech je v základním nastavení několik slabin. Pokud je počítač připojen k síti, zejména do Internetu, administrátor musí dbát na pečlivé dodržování pravidel.

V první řadě se jedná o bezpečnost hesel a vypnutí nepotřebných služeb. Ze statistik vedených útoků vyplývá, že nejčastější metodou je prolamování uživatelských hesel, loginů a přetečení bufferu. Pokud je počítač připojen k modemu hrozí mu metoda „wardialing“. Tento způsob útoku pracuje na hromadném vytáčení telefonních čísel za účelem zjištění, která odpovídají jako modem.

Dále představují bezpečnostní riziko NFS souborové systémy. Špatně nakonfigurovaný systém je velmi náchylný na napadení. Stejně jako NFS je na tom špatně i X Window, který slouží pro vytváření grafického rozhraní. Tato služba by měla být vypnutá, protože obsahuje množství bezpečnostních děr. Mezi další možnosti patří sniffing sítě, kde útočník zachycuje pakety, ze kterých může následně získat důvěrná data. Mezi tyto útoky se také řadí typ útoku zvaný „muž uprostřed“. Existují techniky, které zahlcují konkrétního hostitele IP pakety, zvané „floods“.

Je ještě mnoho dalších technik, před kterými je se třeba bránit a se kterými počítat. Jak se bránit a zajistit, aby nebyla naše síť náchylná k těmto, a dalším útokům bude probráno v kapitole 5 Útok na síť zvenčí.

2 Síťové služby

Síťové služby jsou bezpodmínečnou součástí operačního systému, bez kterého by nebyl tak potřebný jako s nimi. Starají se o spouštění procesů po logování aktivity. Takové služby představují dveře do systému, které je třeba řádně zabezpečit. Pokud se nezabezpečí, pak

jsou tyto dveře pro útočníka otevřeny. Nejlépe zabezpečená služba je vypnutá služba. Ve většině linuxových distribucí je standardně spuštěno maximum služeb s ohledem na maximální pohodlí uživatele. Každé pohodlí s sebou nese i druhou stranu věci - nízkou bezpečnost v podobě nepoužívaných či nechtěných služeb. Protože ne všechny služby jsou v základní konfiguraci nebo bez bezpečnostních záplat bezpečné.

Zjištění a průběžná kontrola dostupných služeb, je dobrou prevencí před průnikem třetí osoby. Kontrola služeb, spouštěných po startu systému a služeb právě běžících, je také nedílnou součástí bezpečnostních prvků.

2.1 Nevhodné služby

Během rozhodování, které služby v systému ponechat, je dobré se řídit následujícími pravidly. Pokud nevíme, co daná služba dělá nebo jí dostatečně nerozumíme, neměla by být zapnutá. Není problém službu dodatečně spustit. Při rozhodování, zdali by měl administrátor nechat puštěnou službu, která představuje určité bezpečnostní riziko nebo službu vypnout s tím, že bude muset oželet její možnosti, měl by volit vždy vyšší bezpečnost. Služby, jejichž bezpečnostní politika je postavena na využívání zdrojových adres UDP, by měl administrátor bezpodmínečně vypnout. Zdrojové adresy se dají velmi snadno falšovat a při povolení těchto služeb by se bezpečnost velmi snížila.

2.2 Nebezpečné služby

Další kategorie služeb, jež je třeba vyřadit ze systému, jsou služby, které neumí přenášet hesla a jiné citlivé údaje v šifrované podobě přes síť. Tyto služby přenáší hesla ve formě prostého textu. Mezi tyto služby patří zejména FTP, telnet, pop3, imap, http a jiné. Přehled velmi nebezpečných služeb, kterých je třeba se vyvarovat je uveden v Tab. 1 - Přehled nebezpečných služeb.

Tab. 1 - Přehled nebezpečných služeb [1]

Stupeň nebezpečí	Název služby	Řešení	Popis
velmi nebezpečné	rsh, rexec	vypnout	rsh spouští příkazy na specifickém zařízení; rexec spouští specifický příkaz na vzdálené stanici
velmi nebezpečné	mount, lockd, statd, quotd	vypnout	mount slouží pro připojení oddílů; lockd blokuje (zamyká) požadavky, které jsou poslány buď lokálně jádrem, nebo vzdáleně jinými procesy; statd spolupracuje s lockd;
velmi nebezpečné	telnet	stelnet, ssh	Slouží k připojení k navázání spojení dvou stanic
nebezpečné	nfs	přepnout nad protokol TCP	nfs slouží ke vzdálenému připojování oddílů; při správné konfiguraci ho lze považovat za bezpečný
nebezpečné	rlogin	řádně nastavit	rlogin umožňuje vzdálené přihlášení; v kombinaci se souborem rhosts lze zvýšit bezpečnost jeho používání
nebezpečné	sendmail	řádně nastavit	poštovní server
nebezpečné	finger	vypnout	Zobrazuje informace o uživatelích na systému
nebezpečné	tftp	vypnout	Zjednodušené FTP
nebezpečné	ftp	sftp	Přenos souborů
nebezpečné	ident (auth)	řádně nastavit	Autentifikace uživatelů skrze TCP
nebezpečné	www (httpd)	https	Služba poskytující běh webových stránek
nebezpečné	pop3, imap	řádně nastavit nebo vypnout	pop3 je protokolem pro elektronickou; imap poskytuje vzdálený přístup do elektronické pošty
představuje určité riziko	systat, netstat	vypnout	netstat zobrazuje aktivitu stanice v síťovém rozhraní (příchozí a odchozí navázané spojení); systat zobrazuje různé statistiky o dané stanici
představuje určité riziko	linuxconf	vypnout	Konfigurace linuxových služeb
představuje určité riziko	named (DNS)	řádně nastavit nebo vypnout	Při zavolání bez argumentů načte základní konfiguraci souboru /etc/named.conf
potenciální hrozba	rwhod	vypnout	Vzdáleně zjišťuje kdo je do systému přihlášen
potenciální hrozba	rwalld	vypnout	Vzdálený uživatel může pomocí rwalld odesílat do systému všesměrové zprávy, které se zobrazí na obrazovkách uživatelů
potenciální hrozba	syslog	řádně nastavit	Zaznamenává zprávy ze systému
potenciální hrozba	talk, ntalk	vypnout	Komunikační program, který kopíruje řádky z terminálu jinému uživateli.
potenciální hrozba	chargen, echo	vypnout	Slouží k testování sítě a systému

2.3 Vhodné služby

Pro zajištění bezpečnosti jakékoliv sítě je nutné vybrat vhodné služby. Vhodné služby se vyznačují dobrou konfigurací a jsou vybaveny šifrovacími protokoly. Pro zvýšení bezpečnosti je dobré udržovat aktuální verze služeb a softwaru, které je možné sledovat na různých emailových konferencích. V těchto zprávách je uvedeno, které bezpečnostní díry byly nalezeny, případně i jak tyto díry odstranit.

Mezi vhodné služby se řadí ty, které podporují SSL/TLS šifrování, jsou dobře nakonfigurované a také ty, které jsou řádně aktualizovány. Pokud používané služby splňují tyto podmínky, dají se označit jako bezpečné.

3 Útoky na síť zevnitř

Ačkoliv útoky mířené na síť jsou nejčastěji vedeny zvenčí, je třeba brát v potaz i to, že se někdo může pokusit napadnout naši síť také zevnitř. Nemusí se jednat přímo o zaměstnance. Může to být i cracker, který prolomil vnější bariéru. Zabezpečením vnitřního sektor, se sníží útočníkovi šance na úspěch.

Během zabezpečování serveru a firemní sítě se musí brát v potaz správné nastavení práv pro skupiny a uživatele. Dalším důležitým faktorem je kontrolovat odkazy na složky či soubory, soubory s uloženými hesly a další, které jsou uvedeny níže.

3.1 Práva

Práva jsou základním kamenem bezpečnostní politika nejen v Linuxu ale ve všech jiných operačních systémech. Není to jen v počítačích, ale i v běžném životě. Privilegia, role a skupiny fungují ve firmách, společnosti a jiných organizacích. Stejně jako člověk nezaměstnaný v dané společnosti nesmí mít přístup do místnosti se servery, tak neoprávněný uživatel nesmí mít například právo zápisu na diskový oddíl.

V systému Linux může existovat mnoho uživatelů a uživatelských skupin. Nejdůležitější a nejmocnější je účet superuživatele (dále jen root). Tento účet je obecně cílem útoku. Pokud útočník získá tento účet, může uhodnout heslo nebo jinak získat dočasné oprávnění roota, je mu celý systém plně k dispozici. Má možnosti upravovat logovací soubory, přidávat bezpečnostní díry a mnoha jinými způsoby maskovat svou aktivitu.

Účet root tedy musí být velmi dobře zabezpečen, aby bylo pro neautorizovanou osobu téměř nemožné dostat se k němu dostat a použít ho. K tomu se musí brát v potaz také přihlášení na terminál. Uživateli root by se měl omezit přístup pouze na vybrané a bezpečné terminály. I ostatní účty je dobré udržovat na určité bezpečnostní hranici. Jedním z vhodných zabezpečení je vytvořit implicitní umask pro všechny uživatele a to buď na hodnotu 027 nebo 022. Pro situace vyžadující maximální bezpečnost se může použít maska 077. Tento příkaz funguje velmi podobně jako chmod akorát čísla vyjadřují, která práva budou odepřena. Aby se předešlo mazání nebo jiným úpravám souborů, ke kterým by útočník neměl mít přístup, postačí nastavit tzv. „sticky bit“. Tento sticky bit má na starosti pouze jediné, aby mohl být daný soubor nebo adresář vymazán pouze svým vlastníkem.

Jedním z velmi mocných příkazů v Linuxu je příkaz „find“. Tento příkaz lze použít například pro vyhledání všech souborů, do kterých mají neomezená práva všichni uživatele na systému.

```
find / ! -fstype proc -perm -2 [1]
```

Příkaz se může upravit tak, aby se vypisoval rovnou do souboru a to následovně:

```
echo | find / ! -fstype proc -perm -2 -ls >> soubor
```

Struktura příkazu je následující[1]:

find / - příkaz prohledává všechny adresáře, mohl by se omezit například na find /home. Zde příkaz vyhledává pouze v adresáři home.

! -fstype proc - zařídí vynechání adresáře /proc, z toho důvodu, že není přítomen fyzicky na disku.

-perm -2 - vybere pouze ty soubory, které mají právo zápisu od všech uživatelů. Argumentem operátoru perm může být „+“, který vybere soubory mající daný bit a operátor bez argumentu, jenž vybírá přesné nastavení uvedených bitů.

-ls - nad každým souborem, který je kontrolován, se provede příkaz ls.

Příkazem find lze také najít špatně nastavené setUID a setGID bity, soubory, které již nemají vlastníky (ti byli ze systému odstraněni) a jiná oprávnění.

Tento proces by měl být spouštěn alespoň jednou měsíčně, čímž se dá odhalit hodně nebezpečných souborů.

3.2 PATH a „.“

PATH je proměnné prostředí, ve kterém jsou uloženy cesty k systémovým adresářům, které bude volaný program prohledávat. Především se jedná o adresáře /bin. Pokud chce uživatel zavolat lokální skript, musí napsat „./nazev_skriptu“ nebo „sh nazev_skriptu“. Aby nemusel psát takto zdlouhavé příkazy, je možné přidat do PATH tečku. Takto sestavené PATH znamená vážnou bezpečnostní díru, která se dá velmi jednoduše zneužít. Útočník může v adresáři vytvořit skript, který bude vykonávat škodlivou činnost. Například takto může útočník získat práva superuživatele.

Obecně může PATH vypadat takto:

```
PATH="/usr/local/bin:/usr/bin:/bin"
```

Špatně nastavená cesta v PATH s tečkou:

```
PATH=".:/usr/local/bin:/usr/bin:/bin"
```

Pokud uživatel spustí „infikovaný“ příkaz v daném adresáři, tak za předpokladu, že tečka v PATH se nachazí na začátku, se spustí nejprve škodlivý kód a poté zadaný příkaz. Infikovaný příkaz se může spustit zcela nepozorovaně. Infikovaný příkaz ls může vypadat následovně[2].

```
#!/bin/sh
#  INFIKOVANY LS
if chmod 666 /etc/passwd > /dev/null 2>&1 ; then
                cp /bin/sh /tmp/.sh
                chmod 4755 /tmp/.sh
fi
exec ls „$@“
```

Pokud by byl tento skript umístěný v adresáři /tmp a root tam spustil příkaz ls, bude /etc/passwd zapisovatelný a zkopíruje se shell pod název .sh do adresáře /tmp.

Toto je jasná ukázka, proč se vyvarovat tečky v PATH. Aby se tomu předešlo, je třeba přidat na konec souboru /etc/profile nebo pro konkrétního uživatele /home/jmeno/.profile tento řádek, který odstraní všechny výskyty „::“ popřípadě tečky:

PATH=`echo $PATH | sed -e 's/::/:/g; s/:.:/:/g; s/:.$//; s/^://';`[2]

3.3 Ochrana hesel

Hesla jsou jednou z nejdůležitějších součástí bezpečnostních prvků v systému. Pokud jsou hesla dostatečně odolná proti útokům, velmi se tím sníží možnost úspěchu útočníka.

Na systémech Linux jsou informace o uživatelských účtech uloženy v souboru /etc/passwd. Tento soubor může vypadat následovně:

```
student:!:100:100:student:/home/student:/usr/bin/sh
guest:*:200:0::/home/guest:/usr/bin/sh
root:x:0:0:system_admin:/root:/bin/sh
```

Jednotlivé položky jsou odděleny dvojtečkou a mají následující význam (jednotlivé položky jsou uvedeny tak, jak za sebou následují v souboru):

uživatelské jméno - používá se pro přihlašování uživatele.

heslo - může obsahovat několik znaků. Nejčastěji používané znaky jsou „x“, „!“ a „*“. X označuje heslo uložené v souboru /etc/shadow. Vykřičník nebo hvězdička značí, že heslo nelze použít (daný uživatel je zablokován).

UID - UserID; všichni uživatelé mají svůj jedinečný identifikátor. Pro roota se používá 0.
GID - GroupID; primární identifikátor skupiny, do které patří uživatel. Informace o skupinách jsou uloženy v souboru /etc/group.

Informace o uživateli - stručné informace o uživateli.

Domácí adresář - absolutní cesta do adresáře, do kterého se uživatel dostane po přihlášení. Pokud není zadán žádný adresář, bude mít uživatel jako domácí adresář kořenový adresář.

Shell - absolutní cesta k shellu. Pro zablokování použijeme /dev/null.

Jelikož je soubor /etc/passwd čitelný pro všechny, značí bezpečnostní díru. Kdyby se hesla ukládala zde, ač v šifrované podobě (dříve to tak bylo), měl by útočník zvýšené šance na úspěch. Tím, že by se dostal k těmto zašifrovaným heslům by se mohl pokusit je rozluštit. Proto se používá tzv. „zastínění hesel“. Hesla se ukládají do /etc/shadow, kam má přístup jen root. Ukázka souboru /etc/shadow:

```
student:$1$fnfffc$pGteyHdicpGOfffXX4ow#5:11012:0:99999:7:::
```

Uživatelské jméno - jméno z /etc/passwd

Heslo - zašifrovaná podoba hesla. „1“ na začátku značí, že jde o algoritmus MD5.

Poslední změna hesla - počet dní, kdy bylo heslo naposled změněno od 1. 1. 1970.

Minimum - počet dní, které zbývají do změny hesla.

Maximum - počet dní, které zbývají do nucené změny hesla.

Varování - počet dní, po které bude uživatel informován před vypršením platnosti jeho hesla.

Neaktivní - počet dní, které zbývají do nucené změny hesla. Pokud uživatel heslo nezmění, bude účet blokován.

Vypršení platnosti - počet dní, odkdy bude účet neaktivní od 1. 1. 1970.

Pro dobré zabezpečení hesel je dobré držet se následujícího schématu:

1) Kvalitní hesla - hesla, jež nejsou snadným cílem slovníkových útoků. Není dobré volit hesla, která s uživatelem úzce souvisí. Nepoužívat všude stejná hesla. Pro tvorbu kvalitních hesel může být použit následující skript.

```
#!/usr/bin/perl
$length;
do{
    print "Zadejte pocet znaku hesla: ";
    chomp($length = <STDIN>);
    print "Heslo musi mit nejmene 6 znaku\n" if $length <=5;
}while $length<=5;
    print generatePassword($length) . "\n";
    exit;
    sub generatePassword
{
#vycet znaku pro generovani
    $possible ='abcdefghijkmnpqrstuvwxyz
                23456789ABCDEFGHJKLMNPQRSTUVWXYZ';
    $special = '!?,.-=+@#$%^&*_';
while (length($password) < $length)
{
    $password .= substr($possible, (int(rand(length($possible)))), 1);
    $password .= substr($special, (int(rand(length($possible)))), 1);
}
    return $password
}
```

2) Zastínění hesel - použití /etc/passwd v kombinaci s /etc/shadow.
3) Vynucení silných hesel - použití vhodných programů jako například passwd+.
4) Omezit platnost - tímto docílíme „násilné“ změny hesla, čímž se útočníkovi omezí čas pro útok hrubou silou.
5) Silné šifrování - pro šifrování hesel používat MD5 algoritmus.
6) Testování hesel - používat programy v pravidelných intervalech pro hádání hesel k ověření jejich bezpečnosti.

3.4 SetUserID, SetGroupID - rizika

SetUID a SetGID jsou velmi dobré nástroje pro usnadnění sdílení souborů. Jedná se o příznak práva, který při spuštění programu jakýmkoliv uživatelem, nastaví programu práva vlastníka nebo skupiny. Příznak SetUID lze nastavit pomocí příkazu chmod u+s a pro skupinu

(SetGID) příkazem chmod g+s. Odebrání se provádí parametrem -s. Následující ukázka značí nastavení souboru pro spuštění s právy vlastníka.

```
$ ls -l /usr/bin/program
-rwxr-xr-x  1 root root 82696 led  4 11:42 /usr/bin/program
$ chmod u+s /usr/bin/program
$ ls -l /usr/bin/program
-rwsr-xr-x  1 root root 82696 led  4 11:42 /usr/bin/program
```

V ukázce je vidět, že místo příznaku „x“ u vlastníka souboru je „s“. Program bude spouštěn s právy roota. Takového příznaku se dá zneužít. Pokud útočník na program zkusí útok přeplněním vyrovnávací paměti, může dostat práva roota.

Bránit se proti takovému útoku lze aktualizováním programů. Pokud program se Set*ID není potřebný, měl by se odinstalovat (více v kapitole 3.1 Nevhodné služby). Pakliže je program potřebný, je třeba nastavit na dané soubory příznak neměnnosti, čímž se zakáže editace souboru. Tento příznak se nastaví pomocí příkazu chattr +i, který může nastavit pouze root. Pro pozdější editaci je třeba soubor zase odblokovat pomocí chattr -i (příkaz chattr je použitelný pouze na souborových systémech ext2, ext3 a ext4). Soubory, které mají nastaven Set*ID lze najít pomocí příkazu find nebo jiných speciálních programů jako je například Nabou.

3.5 Odkazy - symlink, hardlink

Pevný odkaz, „hardlink“, je položka adresáře ukazující na fyzický soubor s daným uzlem. Vytváření pevných odkazů se dělá pomocí příkazu ln. Pokud se smaže zdrojový soubor, neodstraní se fyzicky z disku, jelikož na něj odkazuje ještě pevný odkaz. Pevné odkazy se mohou vytvářet pouze v rámci diskového oddílu. Útočník může zneužít takovýto odkaz a pomocí něj modifikovat nebo smazat obsah souboru nebo pracovat s úplně jiným souborem.

Předejít těmto nástrahám se dá tak, že při instalaci administrátor nainstaluje každý adresář na zvláštní diskový oddíl. Zvlášť by měly být adresáře /home, /var, /tmp, /usr, /boot a kořenový adresář /. Pokud administrátor zajistí, že uživatelé budou moci zapisovat pouze do /home a /tmp, znemožní tím vytváření odkazů do systémových částí.

Symbolický odkaz, „symlink“, je odkaz na soubor. Na rozdíl od pevných odkazů, neukazuje na uzel. Při provádění běžných operací se symbolické odkazy tváří jako cílové soubory. Útočník může takto podstrčit odkazu jiný soubor, než byl původní.

Bránit se proti přepisování a jinému zneužití symbolických odkazů, je možné tak, že pro každý program vytvářející dočasné soubory se budou používat bezpečné funkce, ověřující existenci daného souboru. Pro systémové volání open() k tomu slouží parametr O_EXECL, pro PERL příkaz sysopen a pro shellové skripty mktemp.

4 Útoky na síť zvenčí

Všechny počítače plní svou roli v plném rozsahu, pokud jsou připojeny do internetu. Mohou pak komunikovat s jinými počítači a servery. Jelikož jsou přístupné z internetu, představují ovšem také bezpečnostní riziko, kterému je třeba věnovat pozornost. Útokům zvenčí předchází důkladná příprava a získávání informací o oběti. Existují různé druhy útoků zvenčí. Mezi tyto útoky se řadí sniffing, skenování sítě, mapování portů, zjištění operačního systému, běžících programů ale také získávání informací, které jsou běžně dostupné pro veřejnost. Dále bude uvedeno několik základních typů útoku.

4.1 Mapování sítě

Mapování sítě je jedním z prvních kroků k úspěšnému útoku. Většina serverů poskytuje až příliš mnoho informací o sobě a svých službách. Například při chybové stránce na webserveru se často objevuje typ a verze daného webserveru a jeho přídavné služby, čímž si útočník může zjistit chyby, které by měl zkoušet.

Databáze whois je veřejně dostupná a poskytuje útočníkovi hned několik základních informací jako jméno odpovědné osoby, telefon, adresu a další kontaktní údaje. Tyto informace poslouží jako základní krok k volbě typu útoku, například k sociálnímu inženýrství. Jakmile útočník zjistí IP adresu serveru, může začít zkoumat, které další IP adresy se v síti nachází, k čemuž mu poslouží program nmap. Pakliže najde adresu jako například 95.168.102.26, tak poslední číslo nahradí 0, přidá masku 24 a jednoduchým příkazem si najde zbytek adres:

```
nmap -sP 95.168.102.0/24
```

Parametr -sP zajistil, že program nmap pošle nejen ICMP packet (který může být na firewallu blokován) ale také TCP paket ACK, aby zjistil, zda je host dostupný. Pokud server odpoví paketem RST, čímž dává najevo, že je dostupný.

Další veřejně dostupnou zbraní útočníka je DNS. Zde, pokud se neučiní patřičné kroky, jsou dostupné záznamy o poštovním serveru, jmenném serveru a jiných službách v doméně. K tomu mu poslouží například program host:

```
host -t any domena.cz
```

Pokud bude mít seznam IP adres z dané sítě, může použít reverzní dotazy, kdy pomocí příkazu „host 192.168.2.1“ bude vráceno hostitelské jméno, například ftpserver.domena.cz.

Opatření proti těmto pokusům o zjišťování informací je jednoduché. Nejefektivnější je dávat nicneříkající jména, která útočníkovi neprozradí nic o funkci a významu zařízení. Například místo ftpserver.domena.cz je vhodné pojmenovat zařízení 192.168.2.1.domena.cz.

Skenování otevřených portů a služeb na nich puštěných bude jedním z dalších přípravných kroků útočníka, který podnikne před samotným pokusem o průnik. K tomu bylo vyvinuto hned několik programů. Mezi jedny z nejpoužívanějších patří netcat a nmap. Rozebrán zde bude program nmap, který byl vytvořen nejen ke skenování portů, ale má mnohem větší možnosti použití. Použití programu nmap je popsáno v manuálových stránkách [3]. Nmap je open source nástroj, který umí provádět průzkum sítě a bezpečnostní audit. Nmap využívá surové IP pakety, aby určil hosty, které jsou na síti k dispozici, jaké služby tito hosté nabízejí, běžící operační systémy a mnoho dalších zneužitelných informací. Kromě těchto informací může poskytnout další informace o cílech útočníka, včetně reverzních DNS jmen, operačních systémech, typy zařízení a MAC adresy [4].

Těmto skenováním se dá zabránit příslušným nastavením firewallu a vypnutím nepotřebných služeb.

4.2 Útoky na různé služby

Za velkou bezpečnostní díru je považován systém X Window. Tato služba zprostředkovává grafické rozhraní a používá porty TCP 6000-6063. Jelikož je X Window považován za bezpečnostně nedostatečný, je doporučeno nechat ho na serveru vypnutý. Základní bezpečnostní nástroj pro X Window je příkaz *xhosts*. Při spuštění bez parametrů ukáže seznam všech hostů, kteří mají přístup povolen. Pokud je zadán parametr „+“, znamená to, že je přihlašování povoleno všem stanicím. Parametr *+[nazev_hosta]* přidá daného hosta na seznam počítačů s povoleným připojením. Obdobně se používá odebrání hosta

ze seznamu přes parametr *-[nazev_hosta]*. V případě použití samotného parametru „-“ zapneme přístupový seznam a přístup budou mít pouze ti, kteří jsou na daném seznamu [5].

Zabránit přístupu zvenčí na X Window lze pomocí zablokování daných portů na firewallu. Pokud není X Window nezbytně nutné, je nejlepší volbou jej úplně zakázat.

Není nezvyklé, že některé služby zůstávají v síti v základním nastavení s výchozími (přednastavenými) hesly. Základním pravidlem bezpečnosti je všechna hesla upravit do bezpečné podoby, například s použitím výše zmiňovaného generátoru.

Přehled údajů o síťové konektivitě zprostředkovává program netstat. Tento program pomáhá udržovat přehled o aktivitě systému a zjištění případných podezřelých aktivit. Pravděpodobně bude cílem útočníka, který má superuživatelský přístup, vyměnit ho za falešný. Tento falešný program se chová podobně jako původní, pouze s tím rozdílem, že skryje aktivitu útočníka. Proti takovému falšování (nejen u programu netstat) se dá bránit ověřováním kontrolních součtů u binárních souborů programů. K tomu poslouží například vynikající nástroj TripWire.

Mezi další útoky na služby, které zde budou zmíněny, patří útoky na poštovní servery a webservery. Nejpoužívanějším poštovním serverem je sendmail, kterému bude věnován tento odstavec.

Hlavním problémem poštovních serverů je fakt, že musí být spouštěny pod uživatelem root a to z důvodu připojování na port 25. To znamená riziko napadení serveru a automatického získání superuživatelského účtu. Sendmail má v konfiguračním souboru možnost zvolit volbu „RunAsUser“, která obstará běh programu pod právy zvoleného uživatele. Při této volbě je třeba nastavit příslušná práva na používané soubory pro daného uživatele a skupinu. Jakékoliv změny se uplatní v činnost příkazem

```
killall -HUP sendmail
```

Útok na tento server může předcházet monitorování. Hned po připojení na poštovní server se zobrazí uživateli velmi citlivé informace o poštovním serveru. Abychom takovému průsaku informací, jež mohou pomoci útočníkovi k výběru vhodného útoku, předešli, je třeba modifikovat soubor sendmail.cf a to konkrétní řádek, který obsahuje následující:

```
#SMTP initial login message (old $e marco)
O SmtpGreentingMessage=$j Sendmail $v/$Z; $b
```

Kde název programu „Sendmail“ a proměnnou $v a $Z je třeba změnit:

```
#SMTP initial login message (old $e marco)
O SmtpGreentingMessage=$j Nejaka_matouci_zprava; $b
```

Další informací, kterou může útočník na špatně nastaveném serveru získat je název emailových adres pomocí příkazu VRFY. Tento příkaz slouží k ověřování platnosti poštovní schránky. Stejným způsobem jako pro administrátora může posloužit i útočníkovi. Tuto informaci pak může použít k uhodutí hesla, sociálnímu inženýrství či jinak zneužít. V konfiguračním souboru sendmail.cf je položka #Privacy flags u které se dá příkaz VRFY (a jiné) vypnout. Do položky „O PrivacyOptions=nobodyreturn,authwarnings“ stačí připsat další položku „novrfy“. Podobný význam má příkaz EXPN, který slouží k rozšířeným informacím o poštovní schránce. Vypíná se stejně jako příkaz VRFY,přidáním položky „noexpn“ do konfiguračního souboru. Pro jednodušší zápis do konfiguračního souboru, lze místo noexpn, novrfy (a jiných položek) lze použít „goaway“ [6]. Poštovní server je také mimo jiné třeba chránit před DoS útoky.

Naštěstí lze sendmail chránit omezením zdrojů. Tyto zdroje lze omezit zvláštní konfigurací. Mezi nejběžnější volby patří:

```
(confMAX_DAEMON_CHILDREN)       // omezení maximálního počtu běžících procesů
(confCONNECTION_RATE_THROTTLE) // omezení množství spojení v jedné vteřině
(confMAX_HEADERS_LENGTH)        // omezení velikosti hlavičky pošty
(confMAX_MESSAGE_SIZE)           // omezení velikost zprávy v bytech
```

Dalším článkem na serveru, jenž je potřeba zabezpečit, je webserver. Pokud budeme chtít provozovat webová stránky na vlastním stroji, ne pouze pro intranet, otevíráme tím cestu, kterou může útočník zneužít. Zabezpečení této cesty je tedy nezbytně nutné pro ochránění dat a veškerých interních informací. Nejpoužívanějším webserverem je Apache, který zde bude rozbrán. Každý webserver o sobě implicitně zobrazuje základní informace. Tyto informace jsou obsaženy v hlavičce a stejně jako u sendmailu je třeba tyto informace změnit za falešné. Najdeme je v konfiguračním souboru httpd.h.

Další možností útoku na webserver, je možnost odposlechnutí přihlašovacích údajů. Zamezit takovému útoku lze použitím SSL. Po implementaci SSL bude veškerá komunikace šifrována. Dále je důležité vypnout zobrazování výpisu obsahu složky, což lze provést v konfiguračním souboru Apache direktivou „Option -Indexes“. Velkým bezpečnostním nedostat-

kem trpí CGI skripty, a proto je doporučeno je nepoužívat. Pokud se je třeba je použít, musí se ošetřit a kontrolovat veškeré vstupy.

4.3 Nejčastější metody útoků

Hlavním úkolem útočníka je zjistit, jak je systém jeho oběti zabezpečen. K tomu mu poslouží buď programy, které si napsal sám, nebo programy volně dostupné na internetu. Programem nmap může útočník zjistit otevřené síťové služby, operační systém, otevřené porty a mnohem více, díky čemu může dále postupovat při výběru vhodného útoku.

Hádání a prolamování hesel je další technikou, která je zřejmě nejpopulárnější. Většina uživatelů (a nejen oni ale i administrátoři) podceňují důležitost kvalitního a bezpečného hesla. Pokud se útočník dostane k názvu účtu, emailu nebo jinému přihlašovacímu údaji, může použít útok hrubou silou, kterou se pokusí odhalit heslo. Nejlepší je pro útočníka, když se mu dostane do rukou zašifrovaný soubor s hesly. Pak mu stačí jen spustit příslušný program, například velmi oblíbený JohnTheRipper. Tento program byl konstruován tak, aby rychle a efektivně prolomil hesla založená na nejpoužívanějších algoritmech Blowfish, MD5 a DES. JohnTheRipper si získal velkou oblíbenost díky své rychlosti, spolehlivosti a jednoduchosti. V průběhu zjišťování hesel lze zobrazit stiskem libovolné klávesy aktuální stav programu. JohnTheRipper může běžet hned v několika režimech:

a) **slovníkový útok** - lze definovat seznam slov, která budou použita při prolamování.
b) **brute force útok -** útok hrubou silou, testuje postupně všechny možné kombinace.
c) **externí režim -** poskytuje externí definici režimů pomocí funkcí jazyka C.

Hlavním prvkem opatření proti tomuto útoku je dodržovat zásady tvorby bezpečných hesel. Je doporučováno používat zastíněná hesla a na různých systémech a službách používat různá hesla. Pokud lze, tak na vybraných systémech využít omezené platnosti hesel nebo jednorázová hesla - tyto hesla jsou pro útočníka, pokud se k nim dostane, bezcenná.

Sociální inženýrství je další z nejčastějších technik. Ačkoliv se o ní často mluví a je s důrazem upozorňováno o chránění svých soukromých údajů, existuje stále velká spousta lidí, kteří se nechají těmito triky oklamat. Jedná se o telefonáty, emaily ale dokonce i osobní návštěvy, kde se útočník vydává za někoho jiného. Při falšování identity nemusí jít jen o „živé“ entity ale i o falšování paketů. Při falšování paketů je myšleno například falšování elektronické pošty nebo třeba přímo jednotlivých UDP paketů. Falšování pošty je velmi jednodu-

chá záležitost a je to otázkou pár příkazů, kde napíše útočník falešný email jako odesílatel, za kterého se chce vydávat. Falšovat jde ale i informace jako je ARP tabulka. Tím, že se změní údaje v této tabulce, se může útočník vydávat za jinou osobu a vstoupit tak do navazující relace jako třetí osoba (která tam jinak nemá co dělat). Tomuto druhu útoku se říká Muž uprostřed (anglicky Man-In-The-Middle -> MITM) nebo také ARP spoofing.

Jednou z dalších velmi nebezpečných technik je falšování samotných paketů. Falšovat lze pakety UDP a TCP. Každý paket obsahuje zdrojovou a cílovou IP adresu a port, na kterém bude odeslán. Obsah paketu vytváří ten systém, který ho odesílá, čímž má možnost vytvořit takový paket, který bude mít libovolný (falešný) obsah. UDP pakety se dají snadno zfalšovat tím, že se změní jejich zdrojová adresa. Takový paket se tváří, jako by byl odeslán z vnitřní sítě firmy. Tím se dá prozkoumat například firemní topologie sítě a jiné informace, jež by měly zůstat veřejnosti nedostupné. Zabránit se tomu dá pomocí pravidel firewallu, kde se budou filtrovat pakety vnější sítě, které nesou adresu vnitřní sítě. Takový paket je třeba zanést do logu a následně zahodit. Falšování TCP paketů je o něco složitější proto, že používá speciální algoritmus při komunikaci. Tento algoritmus potvrzuje odesílateli přijaté pakety, a pokud některý z paketů chybí (odesílatel nedostane potvrzení o přijetí) odešle se paket znovu. Při odesílání doplňuje algoritmus do TCP pořadová čísla, jimiž je zajišťována kontrola integrity dat. Průběh navázání spojení TCP komunikace probíhá zjednodušeně tak, že klient zašle serveru zprávu SYN, která nese „počáteční číslo“ (posloupnosti). Server odpoví klientovi zprávou SYN/ACK, v níž uvede své „počáteční číslo“, na kterou odpoví klient zprávou ACK. Pro to, aby mohl útočník tuto komunikaci falšovat, musí znát nebo uhodnout „pořadové číslo“ TCP které je 16bitové (2^{16} možností - šance 1 ku 65536).

Dalším útokem, který může útočník zvolit, je útok na přepínač. Podmínkou tohoto útoku je, že se útočník již v síti LAN nachází. Aby se přepínač choval jako rozbočovač, je třeba pouze zahltit ho pak pakety s různými zdrojovými a cílovými IP/MAC adresami. Poté stačí odfiltrovat vlastní (útočníkovi) pakety a odposlouchávat síť. Aby se zabránilo takovému útoku, může se například rozdělit síť do několika podsítí, které opatříme firewallem.

Jedním z dalších nejčastěji používaných útoků je DoS/DDoS. Základním principem tohoto útoku je vyřadit z činnosti zařízení, které se zahltí požadavky nebo pakety. Tím se daný stroj/služba stává buď nedostupnou, nebo začíná pracovat chybně [7]. Jednou z technik, spadajících do kategorie DoS/DDoS, je takzvaný ping of death. Standardní velikost paketu v protokolu IP je 65535 bajtů. Lze poslat ovšem i paket o větší velikosti tak, že se fragmentuje

a po přijetí, až jej server defragmentuje, dojde k přeplnění paměti, ke zhroucení systému. Dnes je tento „útok“ již nemožný a většina směrovačů tak velké pakety vyfiltruje. Existuje ještě možnost zahltit dotazy PING určitý systém, který potom ve snaze odeslat odpověď je zablokován jak v odchozí, tak v příchozí komunikaci. Další techniky jako packetstorm (paketová bouře) a Smurf, jsou postaveny na podobném principu vyřazení dané služby/systému z činnosti pomocí zahlcením. DDoS útoky jsou v podstatě stejné jako DoS, liší se pouze tím, že nejsou podnikány z jednoho stroje ale z několika najednou, čímž se síla útoku násobí.

Poslední technikou, která zde bude zmíněna, je útok na webový server. Toto je také jedna z nejčastějších technik, díky které bylo již hodně serverů dobyto útočníkem. Především je potřeba vypustit CGI skripty, které jsou ve svém základu velmi špatně zabezpečené. Jednou z možností jak napadnout webserver je SQL nebo PHP injekce. Takový web je možné úplně ovládnout. Pokud je tento server špatně nastaven, může se útočník dostat ke všem souborům na disku, například k souboru */etc/passwd*, či zapisovat jiné soubory na disk. Opatření proti těmto útokům je jednoduché. Je třeba ošetřovat uživatelské vstupy a řádně je filtrovat.

5 Rizika, prevence, náprava průniku

Potencionální riziko narušení bezpečnosti firemní sítě představuje několik faktorů. Prvním faktorem je hlídání si implicitně přednastavených hesel. Tyto hesla tvoří velkou bezpečnostní díru. Stejně jako implicitní hesla, tak ponechané implicitní nastavení služeb nebo systému je dalším faktorem, který snižuje komplexní bezpečnost firmy a jejích údajů. Školení zaměstnanců o bezpečnosti by mělo být nedílnou součástí firemní politiky. Neodborně školený zaměstnanec představuje riziko, díky kterému je ohrožena celá síť. Existuje několik faktorů, které více či méně ovlivňují bezpečnost firemní sítě.

Preventivním opatřením proti narušení bezpečnosti je zejména školení zaměstnanců. Dále je třeba zavést bezpečnostní skripty, které budou automaticky v určité periodě spouštěny. Tyto skripty prohledávají systém, aby našly bezpečnostní nedostatky a informovaly o nich administrátora. Takové skripty si může psát administrátor sám nebo může pomocí volně dostupných programů monitorovat celý systém komplexně. Některé z bezpečnostních programů navíc nabízí i možnost řešení nalezeného problému. Mezi další, poslední zde uvedené, preventivní opatření patří správná konfigurace služeb a používání služeb, které podporují šifrování dat.

Pokud byl systém již napaden, je třeba detekovat veškeré změny v konfiguraci. Takto napadenému systému již není dobré příliš důvěřovat, dokud není potvrzena správná funkčnost. Při zjištění nepříliš závažného průniku do systému lze řešit situaci jednoduše odpojením útočníka ze sítě. Při složitějším napadení je dobré odpojit celý systém od internetu a postupně kontrolovat důvěryhodné služby a hledat služby podvržené.

5.1 Bezpečnostní prvky

Mezi základní bezpečnostní prvky patří aktivní opatření proti útokům. Démon inetd řídí spouštění mnoha služeb na systémech Unix/Linux. Tento démon se dá obalit zvláštním obalem nazvaným TCP wrapper. Ten má na starosti všechna příchozí TCP spojení, která podrobí kontrole. Pokud jsou splněny podmínky, předá TCP spojení patřičné službě. Tyto podmínky se definují v souborech /etc/hosts.deny a /etc/hosts.allow. Běžnou technikou konfigurace těchto souborů je, že se v souboru hosts.deny vše zakáže a pak se teprve povolují služby v hosts.allow. Tyto soubory mohou vypadat například následovně:

```
$ cat /etc/hosts.deny
ALL: ALL
$ cat /etc/hosts.allow
sshd : 192.168.0. EXCEPT 192.168.0.10,192.168.0.44
sendmail: ALL
imapd:  127.0.0.1 192.168.0.0/255.255.255.0
```

Inovací démona tcpd (TCP wrapper) je zavedením xinetd. Tento soubor má v sobě už zakompilováno TCP wrapper, čili se nemusí spouštět navíc tcpd. V tomto démonu je konfigurace mnohem snažší. Obdobou pro zakázání všech služeb jako je tomu v hosts.deny ALL:ALL poslouží příkaz „no_access=0.0.0.0“ nebo lépe „only_from = “. Za rovnítkem se pro zakázání všeho nesmí nacházet nic. Řešení pomocí only_from je mnohem elegantnější, protože lze později přidávat adressy, které se mají povolit. Pro demonstraci konfigurace stejné, jako v předchozím příkladě, bude vypadat konfigurační soubor s xinetd následovně:

```
defaults
{
        instances = 25
        per_source = 5
        log_type = FILE /var/adm/servicelog
        log_on_success = PID HOST EXIT
        flags = NORETRY
        log_on_failure= HOST RECORD ATTEMPT
        only_from = 192.168.0., 127.0.0.1
        #no_access =
```

```
        disabled = nntp uucp tftp bootps who
        shell login exec
        disabled += finger
}



service ssh
{
        socket_type = stream
        protocol = tcp
        wait = no
        user = root
        server = /usr/sbin/sshd
        no_access = 192.168.0.10,192.168.0.44
        access_times = 8:00 - 17:00
}
service sendmail
{
        socket_type = stream
        protocol = tcp
        wait = no
        user = root
        server = /usr/sbin/sshd
        only_from = 0.0.0.0
}
service imap
{
        socket_type = stream
        protocol = tcp
        wait = no
        user = root
        server = /usr/sbin/uw-imapd
        disable = no
        port = 143
        only_from = 127.0.0.1, 192.168.0.0/255.255.255.0
}
```

Jak lze vyčíst z konfiguračního souboru, xinetd má mnoho možností konfigurace jednotlivých služeb. Může omezovat služby na jednotlivé IP adresy nebo jejich rozsahy, omezovat provoz služby pouze na určitý čas, nastavit maximální počet spuštěných instancí pro jednu IP adresu, maximální počet instancí dané služby, logování aktivity a mnoho jiného.

Další možností jak filtrovat spojení na síti je použití nástroje IP tables. Tento nástroj v podstatě hraje roli firewallu (adaptivní, stavový, …). Pravidla při procházení jsou čtena postupně shora dolů. Pokud se nějaké pravidlo vykoná (povolí/zamítne spojení) další se už neberou v potaz. Výchozí politika iptables v Linuxu je „vše povoleno“. Korektní nastavení fi-

rewallu je, že to, co není zakázáno, je povoleno. V tomto případě je dobré držet se hesla „lepší promíjet, než úplně povolit“. Bližší podrobnosti ke kompletnímu ovládání iptables jsou k nalezení v jeho manuálových stránkách. Manuálové stránky lze zobrazit následujícím příkazem.

```
man iptables
```

Velmi dobrým adaptivním firewallem, která vytvořil zkušený linuxový administrátor Bob Toxen, je CrackerTrap. Tento program má mnoho možností v uplatnění. CrackerTrap umí pracovat jak s ipchains tak s iptables. Umí detekovat potencionální útoky a průniky do systému. Jakmile detekuje předdefinovanou akci, během okamžiku zablokuje přístup dané IP adresy do systému a odešle upozornění na daný email, pager, telefon nebo zvukově upozorní na průnik. Tento program se může nakonfigurovat třeba i na sledování určitého portu, který je oblíben u útočníků, a pokud detekuje, že se do něj snaží někdo dostat, okamžitě tato past sklapne - útočník je blokován.

Nedílnou součástí udržení bezpečnosti systému je průběžná kontrola systému. Tato kontrola by se měla skládat z kontroly log souborů a hledání odchylek od normálu. Hledání odchylek spočívá především v nalezení neobvyklých souborů nebo souborů v neobvyklém umístění. Kontrola CRC integrity je neméně důležitá, protože se tak dají odhalit falešné programy, takzvané trojské koně.

V neposlední řadě je vhodné sledovat mailové konference o bezpečnosti. Zejména by mělo být povinností sledovat BuqTraq. Není také od věci navštěvovat různá hackerská fóra, diskusní servery a další podobné stránky.

5.2 Zjišťování škod a náprava

Po zjištění průniku útočníka do systému, je potřeba vyhledat všechny odchylky od normálu. Dobré je začít soubory, které mají pozměněná práva nebo kontrolní součet. Ten označuje jinou verzi programu, pravděpodobně trojského koně. Takovéto soubory se dají vyhledat pomocí příkazu find.

Při zavedení nástroje TripWire před průnikem, můžeme kontrolovat změněné soubory v celém systému. Tento program vytváří kontrolní součet souborů pomocí MD5 algoritmu. Kdykoli pak bude potřeba zjistit, zdali kontrolní součty souhlasí, stačí spustit tento nástroj

a on sám vyhledá a nahlásí případné změny. Vyhledané soubory je třeba nahradit obnovou ze zálohy.

Nebezpečí představují také přenastavení síťových karet do promiskuitního modu. Takové nastavení způsobí, že data nebudou odesílána určitým směrem, ale budou odesílána všesměrově. Při zjištění promiskuitního modu některé ze síťových karet, je třeba před zakázáním zjistit, kdo změnu provedl a začít hledat odchylky od normálu. Pokud by se hned zakázal, mohl by si toho útočník všimnout, zahladit stopy a zmizet. Důležitým krokem je také změnit veškerá hesla v systému.

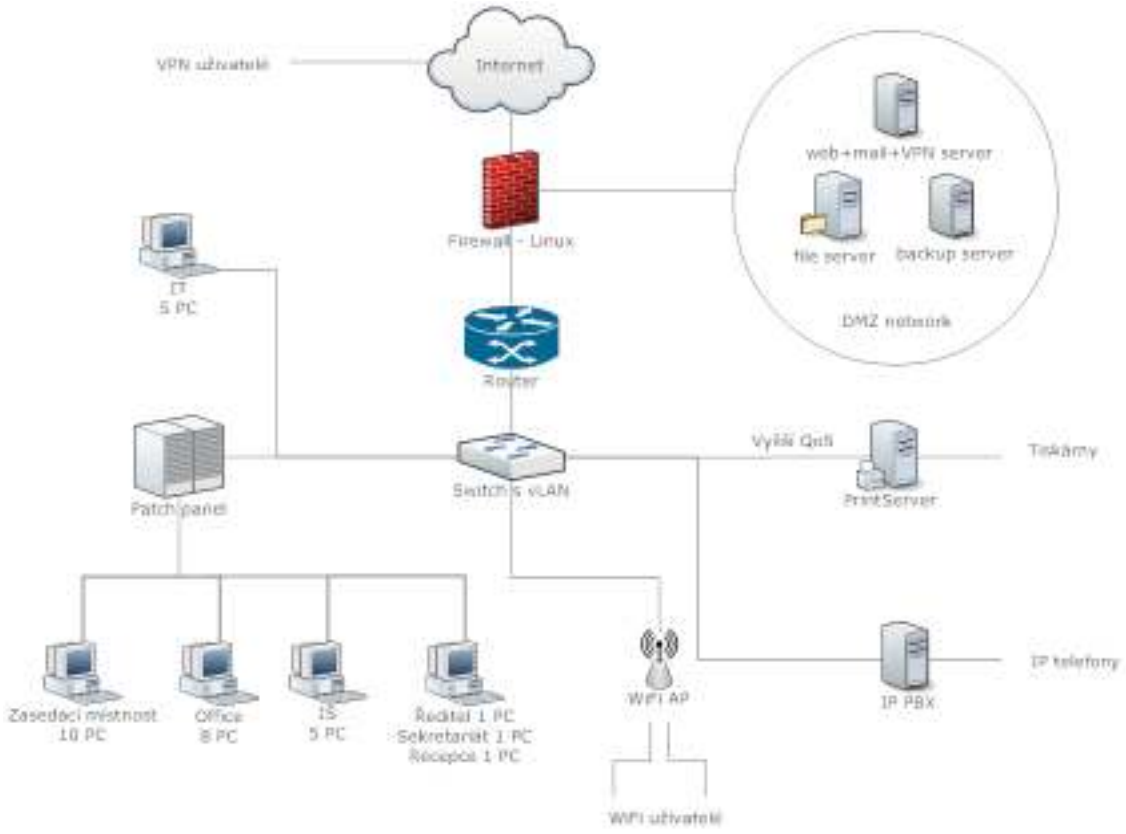
Rozsah škod lze zjistit z většiny logovacích souborů. Tyto soubory sice může útočník, má-li patřičná oprávnění, změnit nicméně zkontrolovat tyto soubory by mělo být povinností.

Veškeré soubory, které zanechal útočník v naší síti, je třeba uložit, pro pozdější důkazy. Nejen tyto soubory ale mohou být důkazem. Jako důkaz také poslouží i neúspěšné pokusy o prolamování do sítě, které je také možno monitorovat.

5.3 Pátráme po útočníkovi

Pokud systém napadne nezkušený útočník, zanechává za sebou víc stop, než si může sám myslet. První chybou útočníka bude, pokud se pokouší prolomit síť ze svého vlastního počítače. Při připojení do systému se loguje IP adresa a někdy také název systému a jeho uživatelské jméno. Pomocí této IP adresy se dá útočník snadno vypátrat. K tomu slouží příkazy nslookup a databáze whois, pomocí kterých lze zjistit, odkud se útočník přihlašoval. Z whois databáze můžeme zjistit i bližší informace jako adresu, telefon a jiné. Cílový systém je třeba kontaktovat a upozornit, že z něj došlo k průlomu do našeho. Toto upozornění je třeba, jelikož cílový systém nemusí být přímo útočníkův, ale jeden z mnoha dalších, napadených systémů, za kterým se útočník maskuje.

6 Firemní síť, server, služby



Obr. 1 - Topologie firemní sítě 1

Firemní síť bude konfigurována na topologii, která je znázorněná na obr. 1. Vstup uživatelů do sítě chrání firewall, který představuje linuxový počítač s iptables. Hned na firewallu se dělí síť na DMZ network (demilitarizovaná síť) a dále na zbytek sítě, která je za směrovačem, jež odděluje vnitřní síť od vnější sítě - internetu. Za směrovačem se nachází přepínač, který pomocí funkce vLAN rozdělí síť na další podsítě. Na těchto podsítích se budou nacházet jednotlivé místnosti, dále pak printserver a tiskárny, IP PBX a telefony a také bezdrátoví uživatelé.

Servery této sítě budou běžet na počítači s linuxem a budou provozovat služby sendmail, která bude obsluhovat emailovou komunikaci, dále Apache, na provozování webserveru a VPN pomocí OpenVPN. Souborový server, bude server provozující uložiště dat a přístup z pracovních stanic s Windows bude řešen pomocí programu Samba. Zálohovací server poběží na Linuxu a budou se na něm ukládat důležitá data. Ostatní stanice budou běžet pod operačním systémem Windows, kromě oddělení IT.

Přehled služeb, které je praktické mít na síti, je uveden v tab. 2. Ne všechny služby budou nainstalovány na firemní síti. Přehled služeb, které budou instalovány, se nachází v příloze B.

Tab. 2 - Služby

Název služby	Popis
Samba	Sdílení souborů pro systémy Windows
Apache	Webserver
Sendmail	Zprostředkovává emailovou komunikaci
OpenVPN	Zprostředkovává VPN spojení
Iptables	Konfigurace firewallu
Arpwatch	Monitoruje arp tabulky a loguje veškeré změny
Ethereal	Monitoruje pakety
Tcp wrapper	Konfigurace firewallu
CrackerTrap	Adaptivní firewall
SSH	Vzdálené připojení
OpenVas	Bezpečnostní software
SARA	Bezpečnostní software
Nmap, netcat	Síťový skener
Snort	Detektor útoku
SHADOW	Monitoruje pakety
JohnTheRipper	Testování síly hesel
PortSentry	Adaptivní firewall
HostSentry	Monitoring podezřelého přihlašování
TripWire	Kontrola pravosti souborů

7 Instalace a konfigurace služeb

Prvním krokem pro instalaci služeb je nainstalování samotného operačního systému Ubuntu v serverové edici. Při instalaci systému je třeba následovat pokynů průvodce. Rozdělení disku na oddíly je třeba udělat následující:

/home
/var
/tmp
/usr
/boot
/

Jedná se o prevenci proti útokům přes hardlink a také proti jiným útokům typu DoS. Uživatelé budou mít nastaven zápis pouze do adresáře /home a /tmp.

Po instalaci systému je třeba provést aktualizace příkazem

```
do-release-upgrade -d
```

kde parametr -d zjistí, jestli je nová distribuce k dispozici. Dále je třeba nainstalovat služby, které budou používány na serveru a na počítači administrátora. Hned po instalaci všech aktualizací je třeba nastavit heslo roota.

```
sudo passwd
```

Pro instalaci služeb a programů slouží následující příkaz:

```
sudo apt-get install nazev_sluzby
```

Příkaz apt-get má mnoho voleb. Pro odinstalování programu stačí nahradit install za remove. Dále často používanou volbou je vyhledání aktualizací balíčku pomocí volby update a upgrade, kde update stáhne informace o nových verzích a upgrade je nainstaluje. Tyto dva příkazy je vhodné spouštět po každé instalaci nového software a poté pravidelně v dané periodě. Pro odinstalování balíčků je třeba vyčistit i balíčky, jež byly svázány s daným balíkem a již nejsou potřeba. Toho lze docílit volbou autoclean. Repozitář dostupných zdrojů balíčku je uložen v souboru /etc/apt/sources.list. Více informací o práci s příkazem apt-get je k nalezení v jeho manuálových stránkách. Další možností jak instalovat úlohy, je použít příkaz *tasksel*, který musí být spouštěn jako root.

Postup instalace a konfigurace bezpečnostních služeb a stanic viz Příloha B.

8 Prvky bezpečnosti - bezpečnostní střípky

Nic nelze stoprocentně zabezpečit, a proto je důležité průběžně hlídat síť a její aktivitu. Specifickou konfigurací a dalšími bezpečnostními opatřeními lze snížit riziko průniku na minimum. Jednou z cest, jak snáze monitorovat aktivitu na síti, je přizpůsobit si /etc/syslog.conf, který nabízí rozsáhlé možnosti monitorování systému.

Školení zaměstnanců a důkladná kontrola všech síťových prvků, zejména nových, je krok správným směrem v ohledu zvyšování bezpečnosti. Každý administrátor by měl být

v ohledu zabezpečení trochu paranoidní. Je lepší později něco povolit, než odstraňovat následky po útoku. Sledování mailových konferencí o bezpečnosti a jiných serverů věnujících se bezpečnosti udržuje administrátorovi vědomosti aktuální.

Závěr, shrnutí

Mým cílem v bakalářské práci bylo zabezpečit a popsat správu menší firmení sítě. Server měl být postaven na platformě Linux a se softwarem podléhajícím licenci GNU/GPL.

Zvolil jsem postup seznámení se zabezpečením sítě a možnostmi jejího narušení. V první řadě jsem definoval a charakterizoval, na co je třeba brát zřetel v rámci zvyšování a následné udržování bezpečnosti. Poté jsem uvedl služby, jež vykazují bezpečnostní nedostatky, a jsou proto nevhodnými či nebezpečnými. Dále jsou popsány služby, které se po správném nastavení dají považovat za bezpečné. V další části jsem poukázal na možné útoky na síť zevnitř i zvenčí a nabídl možnost, jak těmto útokům předejít. V závěru teoretické části jsem uvedl možnosti předcházení útokům, jak zjistit rozsah škod, jak je opravit a jak vypátrat útočníka.

V praktické části jsem se zaměřil na konfiguraci firemní sítě a serverů na platformě Linux. Služby jsem volil dle závěrů v teoretické části a hlavní roli při výběru hrála jejich možnost zabezpečení. Z hlediska použitelnosti a nákladů jsem vybíral software s otevřeným zdrojovým kódem, který je zdarma.

Konfigurace služeb by mohla do budoucna obsahovat komplexnější návod nejen na zabezpečení ale i na samotné jednoduché použití s vlastnostmi, na které je zde pouze odkazováno.

Přínosem mé práce je návod na konfiguraci, zabezpečení a správu menší firemní sítě, postavené na zdarma poskytovaném softwaru. Tato konfigurace zajišťuje vysokou úroveň bezpečnosti a možnost komplexní správy. Tento návod je použitelný pro operační systémy Debian a systémech od něj odvozených. Po drobných úpravách může být použitelný i pro komplexnější sítě. Jedinou placenou částí v tomto manuálu je použitý hardware a operační systém Windows, který je provozován na uživatelských stanicích. Operační systém Windows byl vybrán pro jeho jednoduchou ovladatelnost a proto, že s ním obyčejní uživatelé jsou většinově seznámeni.

Výsledkem této studie je seznámení s pravidly pro bezpečnost, následující kompletním manuálem pro konfiguraci a zabezpečení menší firemní sítě.

Použitá literatura

[1] TOXEN, Bob. *Bezpečnost v Linuxu: Prevence a odvrácení napadení systému.* 1. Brno: Computer Press, 2003. 849 s. ISBN 80-7226-716-7.

[2] HATCH, Brian; LEE, James; KURTZ, George. *Linux - hackerské útoky: Bezpečnost Linuxu - tajemství a řešení*. Praha 8: SoftPress, 2001. 576 s. ISBN 80-86497-17-8.

[3] *Linux Documentation* [online]. 2003 [cit. 2010-03-31]. Nmap(1) - Linux man page. Dostupné z WWW: <http://linux.die.net/man/1/nmap>.

[4] *Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning* [online]. Boston: Nmap Project, 2009 [cit. 2010-03-31]. Chapter 15. Nmap Reference Guide. Dostupné z WWW: <http://nmap.org/book/toc.html>. ISBN 978-0-9799587-1-7.

[5] SCHEIFLER, Bob; GETTYS, Jim. *XFree86 Home to the X Window System* [online]. c1998 [cit. 2010-03-31]. XHOST(1) manual page. Dostupné z WWW: <http://www.xfree86.org/current/xhost.1.html>.

[6] ALLMAN, Eric; SHAPIRO, Gregory Neill; ASSMANN, Claus. *Sendmail Operation Guide* [online]. 2004 [cit. 2010-03-31]. Dostupné z WWW: <http://freenet-homepage.de/slgig/op_en/options.html>.

[7] *CERT/CC* [online]. 1997 [cit. 2010-03-31]. Denial of Service. Dostupné z WWW: <http://www.cert.org/tech_tips/denial_of_service.html>.

[8] HA, John, et al. *Red Hat Linux: Security Guide* [online]. North Carolina: Red Hat, Inc., 2002 [cit. 2010-04-08]. Dostupné z WWW: <http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/pdf/rhl-sg-en-9.pdf>.

[9] *Linux Documentation* [online]. 2003 [cit. 2010-03-31]. rdiff-backup(1) - Linux man page. Dostupné z WWW: <http://linux.die.net/man/1/rdiff-backup>

[10] FALKO, Timme. Automated Backups With rdiff-backup. *HowToForge* [online]. 2005, 1, [cit. 2010-04-13]. Dostupný z WWW: <http://www.howtoforge.com/linux_rdiff_backup>.

[11] RODDINS, Steven. OpenVPN: Easy and Secure Setup Guide. *Documents* [online]. 2010-01-06, 1, [cit. 2010-05-04]. Dostupný z WWW: <http://stevenroddis.com/documents/OpenVPN_Easy_and_Secure_Setup_Guide.pdf>

Příloha A - Slovníček pojmů

ARP - Adress Resolution Protocol, používá se pro získání MAC adresy z jeho IP adresy.

ARP spoofing - falšování ARP tabulky.

Bezpečnostní díra - chyba v systému nebo zdrojovém kódu programu, která představuje riziko zneužití neoprávněnou osobou.

Cracker - člověk s vysokými znalostmi v oblasti IT, často programátor, jenž se nabourává do systému za účelem poškození nebo zneužití dat. Často bývá zaměňován s pojmem „hacker“ označujícím jedince, který, na rozdíl od crackera, na chyby upozorňuje a pomáhá je řešit.

DMZ - DeMilitarized Zone. Oblast v síti, která si vyžaduje speciální bezpečnost, jež je zaručena separováním ze sítě. Veškerý její přístup by měl chránit firewall.

DoS - Denail Of Service, útok, jehož cílem je vyřadit cílový stroj z provozu.

GNU/GPL - licence, která opravňuje k volnému používání, šíření zdarma nebo i za úplatu a jiné poskytování produktu. Toto poskytování má jediné omezení a to takové, že produkt stále spadá pod GNU/GPL licenci.

HMAC - Zpráva autentizačního kódu, který používá pro výpočet kryptografické funkce s tajným klíčem.

Chmod - změna oprávnění souborů.

IP PBX - také VoIP, je technologie, která přenáší hlas v digitalizované podobě pomocí protokolů UDP/TCP a IP prostřednictvím počítačové sítě.

Logovací soubor - soubor, který zobrazuje akce, jež se udály.

MAC adresa - hardwarová adresa, je jednoznačným identifikátorem síťového zařízení. U moderních zařízení lze změnit. Pracuje na spojové vrstvě OSI modelu.

Open source - Licence software s otevřeným zdrojovým kódem.

Slovníkový útok - metoda útoku na uživatelské účty, kdy útočník použije soubor s několika (až stovkami) frázemi, která se automaticky zkouší jako heslo k uživatelskému jménu.

SSL/TLS - protokol, který umožňuje navázat šifrované (bezpečné) spojení mezi koncovým uživatelem a serverem.

TCP - Transmission Control Protocol, spojovaný komunikačni protokol, kontroluje integritu odeslaných a přijatých dat.

UDP - User Datagram Protocol, používá nespojovanou komunikaci, při které se nezajímá o doručení paketu.

Umask - příkaz umask nastavuje implicitní práva na vytvořeném souboru.

Útok hrubou silou - metoda útoku, kterou útočník zkouší všechny možné kombinace znaků. Tato metoda je časově velmi náročná a u dlouhých a silných hesel v reálném čase nerealizovatelná.

vLan - virtual Local Area Network, virtualní vnitřní síť. Jedná se o funkci, kdy přepínač rozdělí síť na jednotlivé segmenty, které na sebe nevidí.

Zastíněná hesla - uživatelská hesla, uložená v zašifrované podobě v souboru /etc/shadow.

Příloha B - konfigurace služeb

Konfigurace webserveru

Pro webserver je nejvhodnějším softwarem Apache, díky své vysoké bezpečnosti. Apache je nejrozšířenějším software s otevřeným zdrojovým kódem na světě a je zdarma. Webserver se nainstaluje níže uvedeným příkazem.

```
$ sudo tasksel
```

V tabulce, která se objeví po zadání příkazu, je třeba vybrat LAMP server. Tento balíček obsahuje PHP, MySQL a Apache. Pro instalaci samotného Apache by byl použit příkaz

```
$ sudo apt-get install apache2
```

Po kontrole, zda nejsou další aktualizace k dispozici, je třeba Apache správně nakonfigurovat. Většina konfiguračních nastavení se provádí v souborech httpd.conf a apache2.conf. Apache musí být spouštěn pod právy superuživatele, jelikož navazuje spojení na portu 80. Veškeré porty do portu 1024 potřebují práva superuživatele při navazování spojení. Po navázání spojení je ale třeba změnit uživatele, pod kterým Apache poběží, proto na řádcích v konfiguračním souboru změníme Group a User uživatele na libovolného. Tyto řádky mohou vypadat následovně:

```
User web
Group web
```

Kořenový adresář by měl mít nastaven přístupový mod 755

```
$ chmod -R 755 /var/www/
```

a protokolové soubory by měly mít mod 600 a vlastnit by je měl pouze superuživatel.

```
$ sudo chown root /var/log/apache2/*
$ sudo chmod 600 /var/log/apache2/*
```

Z důvodu předcházení napadení zevnitř, je třeba zakázat vytváření souborů .htaccess které mohou měnit bezpečnostní direktivy v hlavním konfiguračním souboru. Následující kód musí být umístěn před direktivy adresářů a demonstruje zakázání tvorby souboru .htaccess.

```
<Directory />
AllowOverride None
```

```
Options None
allow from all
</Directory>
```

Takto lze nastavit i přístup do jednotlivých adresářů pro Apache a to i pro jednotlivé IP adresy.

```
<Directory /cesta/k/adresari/>
order deny,allow
deny from all
allow from 192.168.0.0/16
</Directory>
```

Nebo je možné zablokovat i konkrétní IP adresu nebo celý rozsah adres, kde v direktivě „deny from“ se zadá místo all daná IP adresa.

Pomocí direktivy FilesMatch je možné změnit přístup k souborům, která se zapisuje za direktivy, operující s adresářemi. Pro zakáz zobrazování souboru admin.php poslouží níže uvedený kód.

```
<FilesMatch „cesta/k/souboru/admin.php“>
Deny from all
</FilesMatch>
```

Lze také nastavit pro vybrané soubory jako je .php, .html, .htm kódování a jazyk.

```
<FilesMatch „\.(htm|php|html)$“>
AddDefaultCharset UTF-8
DefaultLanguage cs-CZ
</FilesMatch>
```

Pro útočníka může být užitečné, pokud se bude vypisovat obsah adresáře, kde není index soubor. Toto chování se potlačí direktivou Options -Indexes.

Je třeba také změnit hlavičku, jež je poskytována pro veřejnost. Útočník se z ní může dozvědět verzi a typ serveru. Z důvodu vyšší bezpečnosti je tedy doporučeno změnit v souboru /src/include/httpd.h řádky

```
#define SERVER_BASEPRODUCT „Apache“
#define SERVER_BASEREVISION „2.2.15“
```

na libovolné, například takto:

```
#define SERVER_BASEPRODUCT „Open source“
#define SERVER_BASEREVISION „1.0“
```

Na serveru nebudou běžet CGI skripty, které bez hlubší kontroly představují vážné bezpečnostní riziko. Pro restart služby apache slouží příkaz

```
$ sudo /etc/init.d/apache2 restart
```

Konfigurace poštovního serveru

Sendmail je nejrozšířenějším emailovým serverem s otevřeným zdrojovým kódem a je pro veškeré použití zdarma. Díky otevřenému zdrojovému kódu je jeho bezpečnostní podpora velmi vysoká. Otevřený kód ovšem představuje dvousečnou sekeru, protože útočník může hledat slabiny v kódu programu. I přesto byl zvolen sendmail jako emailový server pro konfiguraci do firemní sítě. Instalace sendmailu se provede příkazem

```
$ sudo apt-get install sendmail
```

Po instalaci zkontrolujeme, zda nejsou dostupné nějaké aktualizace jako u webserveru. Je třeba nastavit na konfigurační soubor sendmailu příslušná práva

```
$ sudo chmod 744 /etc/mail/sendmail.cf
```

V sendmailu se nachází hned několik příkazů, které by měli být blokovány z důvodu navýšení bezpečnosti. Mezi tyto příkazy patří VRFY, který ověřuje platnost emailové adresy a EXPN, který má podobné chování jako VRFY. Dále je pak třeba zablokovat oznamování obsahu fronty zpráv pomocí direktivy „restrictmailq“ a vyžádání takzvaného pozdravu, při navazování spojení. Tento „pozdrav“ je v podstatě oznámení hostitelského jména a IP adresy. K zavedení pozdravu slouží direktiva „needmailhello“. Kvůli vyloučení použití služby DoS je třeba nastavit také maximální počet potomků programu direktivou MaxDaemonChildren. Dalším krokem ke zvýšení bezpečnosti systému je zakázat zobrazování informací o sendmailu. Ten sám o sobě při navázání spojení oznámí svou verzi. Kompletní změněný kód vypadá následovně:

```
define(`confPRIVACY_FLAGS',needmailhelo needexpnhelo needvrfyhelo restrictqrun
        restrictexpand nobodyreturn authwarnings restrictmailq novrfy noexpn)dnl
define(`confSMTP_LOGIN_MSG',$j Sendmail -Secured)dnl
define(`confMAX_DAEMON_CHILDREN', 25)dnl
```

Po provedení úprav v konfiguračním souboru /etc/sendmail.mc je třeba vždy zavést změny do hlavní konfigurace sendmailu příkazem

```
$ sudo sh -c “sudo m4 /etc/mail/sendmail.mc > /etc/mail/sendmail.cf“
```

a následně restartovat službu

```
/etc/init.d/sendmail restart
```

Ověření vypnutí služeb lze následujícími příkazy:

```
$ telnet mail.company.com smtp
220 mail.company.com ESMTP Sendmail -Secured
$ vrfy root
252 2.5.2 Cannot VRFY user; try RCPT to attempt delivery (or try finger)
$ expn root
502 5.7.0 Sorry, we do not allow this operation
$ quit
221 2.0.0 mail.company.com closing connection
Connection closed by foreign host.
```

Pomocí direktivy FEATURE(`relay_entire_domain'), v konfiguračním souboru /etc/mail/sendmail.mc, povolíme řízené přeposílání pošty. Další soubor, pomocí kterého lze řídit přijímání nebo odesílání pošty je soubor /etc/mail/access. V tomto souboru se dá přesně řídit práce s poštou. Následující kód ukazuje, jak zakázat veškeré emaily z domény hotmail.com ale účet info@hotmail.com povolit, Dále bude zakazovat poštu z kteréhokoliv počítače v sítí 88.126.*.* a účtům začínajícím spam@... bude vracet chybové hlášení „No spammer allowed“. V doméně company.com bude povoleno přeposílání pošty.

```
spam@              ERROR: "550 No spammer allowed"
hotmail.com        REJECT
info@hotmail.com   OK
88.126             REJECT
company.com        RELAY
```

Veškeré změny je třeba zapsat do databáze přístupů v souboru /etc/mail/access.db příkazem

```
$ sudo sh -c “sudo makemap hash /etc/mail/access < /etc/mail/access“
```

a restartovat sendmail.

Podobně lze i filtrovat odesílání pošty pomocí direktivy FEATURE('blacklist_recipients') v souboru /etc/mail/sendmail.mc. Opět je třeba zavést změny do konfigurace a restartovat službu. Pak už lze jen přidávat do souboru /etc/mail/access příslušné řádky. Například pokud má účetní dovolenou nebo je účet pepa@is.company.com dočasně zablokován, lze nastavit chybové zprávy s pomocí příkazů:

```
ucetni                  ERROR: “550 Ucetni je na dovolene.“
pepa@is.company.com     ERROR: “550 Tento ucet byl docasne pozastaven“
```

Jako obranu proti DoS útoku, který by zahltil diskovou kapacitu, je třeba omezit kapacitu jednotlivým uživatelům na rozumné množství pošty. Lepším řešením je nastavit omezení pro celou skupinu mail, která pracuje s odesíláním pošty. Zapsáním položky „usrquota“ na čtvrté pole k diskovému oddílu /var bude aktivována disková kapacita. Dále je třeba vytvořit dva soubory na oddílu

```
$ touch /var/quota.user
$ touch /var/quota.group
```

a nastavit oboum mod 600. Kvótu pro skupinu mail lze potom nastavit příkazem

```
$ edquota -g mail
```

kde se nastaví požadované změny. Pro nastavení kvóty jednotlivým uživatelům slouží parametr -u.

Pro zjištění nejvíce obsazených schránek slouží následující příkaz, který navíc odešle mail uživateli root. Aby se mohl email odeslat, je třeba nainstalovat balíček „mailutils“. Tento příkaz je dobré spouštět automaticky pomocí služby cron.

```
$ ls -sS /var/spool/mail | head -20 | mail -s preplnene_mailboxy root
```

Konfigurace VPN serveru

VPN server bude jednou z klíčových oblastí, kde je třeba se zaměřit na bezpečnost. VPN obstarává připojení zvenčí do firemní. Vybrán byl produkt OpenVPN podléhající licenci GNU/GNL. Z důvodu otevřeného zdrojového kódu je jeho bezpečnost často testována. Umožňuje navázat spojení pomocí certifikátu, klíče nebo na základě autentizace. Instalace se provede příkazem apt-get. Spolu s OpenVPN je třeba nainstalovat ještě balík ssh a openssl.

```
$ sudo apt-get install ssh
$ sudo apt-get install openssl
$ sudo apt-get install openvpn
```

Po instalaci těchto balíčků je třeba se přihlásit pod účtem root pomocí příkazu su. Do souboru /etc/default/openvpn se přidá na konec souboru direktiva, která umožní jednodušší konfiguraci.

```
AUTOSTART=“openvpn“
```

Dále je třeba zkopírovat složku /usr/share/doc/openvpn/examples/easy-rsa/ i s jejím obsahem do složky /etc/openvpn/.

```
$ cp -r /usr/share/doc/openvpn/examples/easy-rsa/ /etc/openvpn/
```

Po zkopírování se musí upravit soubor vars, ve kterém se nachází údaje o serveru a klíčové informace.

```
$ vim /etc/openvpn/easy-rsa/2.0/vars
```

Editovat je třeba následující řádky na konci souboru. Do emailové adresy není doporučeno uvést přímý email roota ale nějaký zvláštní pro VPN.

```
export KEY_COUNTRY="CZ"
export KEY_PROVINCE="CZ"
export KEY_CITY="Pardubice"
export KEY_ORG="Company"
export KEY_EMAIL="vpnadmin@company.com"
```

Jakmile je soubor editován je třeba spustit soubor vars a vyčistit konfiguraci.

```
$ . ./vars
$ ./clean-all
```

Další částí konfigurace je vytvoření certifikátu serveru a vlastní certifikační autority. Veškeré dotazy se nechají v předvoleném nastavení. Pouze sekce se nastaví na ITSec. U druhého příkazu necháme passphrase prázdné a „An optional company name“ také. Na poslední dvě otázky je třeba odpovědět „y“ (ano).

```
$ ./build-ca
$ ./build-key-server server
```

Je třeba také vytvořit certifikát pro klienta, kde se zadá opět ITSec jako sekce a zbytek se nechá ve standardním nastavení. Na konci se poslední dvě otázky potvrdí. (Pro více klientů je třeba vytvořit nové certifikáty. U příkazu je třeba změnit pouze parametr client1 na client2.)

```
$ ./build-key client1
```

Dokončení konfigurace se potvrdí příkazem

```
$ ./build-dh
```

Nyní je třeba vygenerovat TLS-auth klíč ke zvýšení bezpečnosti, který používá při navazování spojení HMAC. Zvyšuje bezpečnost vůči DoS útokům, paketovým bouřím, skenování portů a dalším typům

```
$ openvpn --genkey --secret /etc/openvpn/ta.key
```

Konfigurace klienta je velmi jednoduchá. Stačí pouze zkopírovat soubory ca.crt, clientX.crt a clientX.key, kde X je číslo klienta (v konkrétním případě client1.crt a client2.key), ze složky /etc/openvpn/easy-rsa/2.0/keys/ a překopírovat je na klientskou stanici. Všem klientům je pak třeba ještě přidat soubor ta.key z /etc/openvpn/.

Jako další krok konfigurace je třeba ještě nastavit překlad adres (NAT). V souboru sysctl.conf je třeba povolit (odkomentovat) řádek pro směrování IP adres, oznámit toto nastavení souboru ip_forward a nastavit pravidlo na firewallu (více v Příloze B).

```
net.ipv4.ip_forward=1
$ echo 1 > /proc/sys/net/ipv4/ip_forward
```

Na serveru se nastaví konfigurační soubor openvpn.conf, který je třeba vytvořit a to následujícím způsobem. [11]

```
$ vim /etc/openvpn/openvpn.conf
```

Do toho souboru se vloží následující řádky.

```
dev tun
proto udp
port 443
ca /etc/openvpn/easy-rsa/2.0/keys/ca.crt
cert /etc/openvpn/easy-rsa/2.0/keys/server.crt
key /etc/openvpn/easy-rsa/2.0/keys/server.key
dh /etc/openvpn/easy-rsa/2.0/keys/dh1024.pem
tls-auth ta.key 0
cipher AES-256-CBC
user nobody
group nogroup
server 10.3.0.0 255.255.255.0
persist-key
persist-tun
cipher AES-256-CBC
client-to-client
push "dhcp-option DNS 208.67.222.222"
push "dhcp-option DNS 208.67.220.220"
```

Konfigurace klienta probíhá instalací OpenVPN[1] a zkopírováním vytvořených souborů na serveru do složky C:\Program Files\OpenVPN\config. Soubory, které je třeba zkopírovat, jsou následující.

```
ca.crt
client1.crt
client1.key
ta.key
```

Do stejné složky je třeba vytvořit ještě konfigurační soubor external.ovpn s obsahem uvedeným níže.

```
dev tun
client
proto udp
remote XXX.XXX.XXX.XXX 443 # Doplnit vnejsi IP adresu firmy
resolv-retry infinite
nobind
user nobody
group nogroup
redirect-gateway def1
persist-key
persist-tun
ca ca.crt
cert client1.crt
key client1.key
tls-auth ta.key 1
cipher AES-256-CBC
remote-cert-tls server
verb 3
```

Použití konfiguračního souboru external.ovpn slouží k přípojení zvenčí sítě. Po nastavení konfiguračního souboru pak jen stačí pustit klienta, který se zobrazí na panelu Start vpravo. Samotné připojení se provede kliknutím pravým tlačítkem na ikonu a vybere se akce Connect.

Konfigurace souborového serveru

Jako souborový server byl vybrán systém Samba. Je to nejlepší kandidát na sdílení souborů a tiskáren mezi systémy Linux a Windows. Samba je systém spadající pod licenci GNU/GPL a je zdarma. Bezpečnost je zajištěna četnými aktualizacemi a bezpečnostními revizemi. Instalace serveru se provede příkazem

[1] OpenVPN klienta pro Windows lze stáhnout na http://openvpn.net/index.php/open-source/downloads.html

```
$ sudo apt-get install samba
```

nebo pomocí tabulky předpřipravených balíčku, kde se vybere „Samba fileserver“.

```
$ sudo tasksel
```

Hlavním konfiguračním souborem je zde soubor /etc/samba/smb.conf, odkud se řidí činnost celého systému. Oprávnění tohoto souboru musí být maximálně 644, což by mělo být standardní nastavení. Stejně jako tento soubor by měl mít možnost modifikovat pouze root, tak by měl být i jeho vlastníkem pouze on.

Správa uživatelů se provádí pomocí příkazu smbpasswd. Pro přidání uživatele se používá parametr -a (append) a jméno uživatele použité v linuxovém stroji. Pro odebrání účtu slouží parametr -d (delete). Velmi důležité je dávat pozor, aby při vymazání uživatele ze systému byl vymazán uživatel i ze Samby. Jinak vzniká bezpečnostní díra. Při vkládání nových uživatelů je dobré dávat je všechny do jedné skupiny, pro kterou platí jednoznačná oprávnění. Každý uživatel, který je přidáván do systému Samba, musí mít záznam v souboru /etc/passwd, což znamená, že musí mít účet na linuxovém stroji.

Běžným příkladem vytvoření uživatele by mohl být následující sled příkazů

```
$ useradd -g users -d /dev/null -s /dev/null -n -u 500 uzivatel
$ smbpasswd -a uzivatel
```

Tento uživatel nebude mít žádný domovský adresář, ani žádný implicitně přednastavený příkazový interpret. Smí pouze přistupovat ke sdíleným objektům Samby. Pokud by měl uživatel mít i svůj vlastní domovský adresář, je třeba změnit část s parametrem -d na /home/uzivatel/ a nastavit mod na 700. Přístup uživatelů lze omezit v direktivě [global] na omezení přístupu využívající soubory /etc/passwd a smbpasswd.

```
[global]
        security = user
```

Existují ještě další hodnoty direktivy security. Zejména se jedná o share a domain nebo server. První, share, není bezpečná, a proto není doporučována. Druhá, domain/server, bude stavět server Samba do role doménového serveru, kde se uživatelé při přihlašovaní budou muset přihlašovat do dané domény.

Protokolování systému se provádí do speciálních souborů v adresáři /var/log/samba/. Toto lze změnit v konfiguračním souboru, kde se dá nastavit nejen cesta ale i parametry protokolu.

```
log file = /var/log/samba/mainsmb.log
max log size = 50
```

Právo modifikovat logovací soubory by měl mít pouze root a měl by být i jejich vlastníkem. Při testování systému Samba je dobré použít direktivu debug level, zapsanou do konfiguračního souboru smb.conf.

```
debug level = 0
```

Tato direktiva má několik úrovní, které označuji obsáhlost výpisu. Základní úrovně jsou 0-3, kde 3 je nejvíce rozepsaný výpis a 0 je výpis úplně vypnutý, Protokolovacích úrovní je mnohem více, ale tyto už nejsou třeba. Při ladění systému je dobré mít zapnutý obsáhlý výpis, jinak by měl být debug level roven 0.

Při spuštění Samby se dají do provozu dva démoni: smbd a nmbd. Tito pracují na portech 137, 138, 139 a 445, které je třeba na firewallu zablokovat zvenčí - více v Konfigurace firewallu.

Základním bezpečnostím opatřením při konfiguraci Samby je pravidlo nesdílet kořenový adresář. Již při rozdělování disku byl vyčleněn oddíl pro /home a právě zde budou umístěny skupiny uživatelů v Sambě. Zde je třeba vytvořit složku /samba a do ní adresáře /private a /public. Private adresáři se nastaví přistupový mod 2770 a veřejnému 2775 a změní se jim skupina. Private bude mít skupinu private a public bude pod skupinou users. Toto se provede následujícími přikazy:

```
$ sudo mkdir /home/samba /home/samba/private /home/samba/public
$ sudo chmod 2775 /home/samba/public
$ sudo chmod 2770 /home/samba/private
$ groupadd private
$ sudo chgrp private /home/samba/private
$ sudo chgrp users /home/samba/public
```

Tyto složky se vytvářely proto, že veškerý přístup k nim bude chráněn skupinovým oprávněním. Aby se zajistil odpovídající chod a přístup k souborům, musí se upravit ještě konfigurační soubor smb.conf a to následujícím způsobem:

```
[public]
        path = /home/samba/public
        public = yes
        writeable = yes
        directory mask = 0755
        create mask = 0775
        hide dot files = true
[private]
```

```
path = /home/samba/private
public = no
writeable = yes
directory mask = 0771
create mask = 0771
hide dot files = true
force group = @private
valid users = @private, jitka.office, michal.is, david.is
write list = jitka.office, david.is
read list = michal.is
```

Další adresáře lze vytvářet například i pro zálohy souborů, ke kterým by neměl mít nikdo přístup nebo soukromé projekty určitého oddělení, pro které se opět vytvoří daná skupina a pomocí direktivy valid users povolí konkrétním uživatelům přístup.

Podobně se dá nastavit sdílení tiskáren na síti pomocí následujících direktiv.

```
[print$]
   path = /var/lib/samba/printers
   browseable = yes
   guest ok = yes
   read only = yes
   write list = root
   create mask = 0664
   directory mask = 0775

[printers]
   path = /tmp
   printable = yes
   guest ok = yes
   browseable = no
```

Dle předchozí úpravy konfiguračního souboru je třeba upravit ještě soubor /etc/group kam ke skupině private zapíšeme následující uživatele. Příslušný řádek bude vypadat následovně.

```
private::10000:jitka.office,michal.is,david.is
```

Díky tomu se do adresáře /home/samba/private, s přistupovým módem 770 nedostane nikdo, kdo není členem skupiny private.

Konfigurace zálohovacího serveru

Zálohovací server slouží k ukládání důležitých dat, které je třeba schraňovat kvůli pozdějšímu použití nebo pojištění proti ztrátě. Na tento server musí mít přístup pouze root. Pro obsluhu a koordinaci záloh je nainstalován program rdiff-backup, který využívá program

rsync a obohacuje ho o další vlastnosti. Při zálohování používá speciální algoritmus, který kontroluje pouze změny v zálohovaném objektu a ty přenáší, čímž se výrazně zrychluje zálohovací čas. Mezi hlavní přednosti programu rdiff-backup patří, že umí zálohovat nejen lokálně ale i po síti s pomocí ssh. Rdiff-backup zachovává práva, umožňuje obnovit i smazané soubory a v neposlední řadě vede záznamy o zálohách.

Program rdiff-backup je ve standardních repozitářích, a proto k instalaci postačí pouze následující příkaz, který se spustí na souborovém i zálohovacím serveru.

```
$ sudo apt-get install rdiff-backup
```

Vynikajícího zálohovací procesu lze dosáhnout pomocí kombinace rdiff-backup a automatického plánovače cron. Takto bude zajištěna automatická záloha a postačí ji pak jen občasně zkontrolovat, zdali vše probíhá jak má.

Základní příkaz pro zálohování má následující syntaxi

```
$ rdiff-backup /home/adresar /mnt/backup/adresar
```

kde /home/adresar je adresář, který má být zálohován a druhá cesta ukazuje, kam bude zálohován. Možnosti programu jsou ale mnohem větší. Lze například vyčlenit adresáře či soubory, které se zálohovat nemají nebo naopak mají. K tomu slouží parametry --exclude a --include.

```
$ rdiff-backup /home/adresar /mnt/backup/adresar --exclude /home/adresar/docs \
--include /home/adresar/docs/firma
```

Takto se nebudou zálohovat soubory v adresáři docs, ale firemní dokumenty se zazálohují. Příkaz je oddělen zpětným lomítkem, ale jinak musí být zapsán na jednom řádku. Pokud administrátor neví jistě, zdali jsou v názvu velká či malá písmena, může použít direktivu ignorecase. K rozšíření nástroje include/exclude lze použít také regulární výrazy.

```
$ rdiff-backup /home/adresar /mnt/backup/adresar --exclude /home/adresar/docs \
--include ignorecase:'/home/adresar/docs/firma'
```

Zálohovat lze i po síti, kde je třeba pouze uvést plnou síťovou cestu a uživatele pod kterým budeme provádět operaci.

```
$ rdiff-backup backup@sambaserver.company.com::/home/samba/michal \
backup@backupserver.company.com::/home/backup/is/michal
```

Tímto příkazem se zálohují veškeré soubory uživatele Michal, které se nachází na Samba serveru. Při zadávání toho příkazu je samozřejmě nutné se autorizovat.

Obnova souborů probíhá stejně jen s parametrem -r a časovým údajem ze kdy chceme obnovit zálohu. Prvním zdrojem je cesta k záloze a druhá cesta ukazuje kam zálohu přemístit. Následující příkaz obnoví zálohu vytvořenou před dvěma týdny.

```
$ rdiff-backup -r 2W /mnt/backup/adresar /home/adresar/recover
```

Z důvodu úspory místa je dobré mazat zálohy starší než například měsíc. K tomu slouží parametr --remove-older-than za který se přidává časový údaj, stejně jako u obnovy souborů a parametr --force.

```
$ rdiff-backup --remove-older-than 1M --force /mnt/backup/adresar
```

Pokud existuje pouze jediná záloha, která je starší než měsíc, nesmaže se, protože --remove-older-than se týká více verzí záloh. U této direktivy se dá použit ještě omezení na maximální počet záloh, kdy se určí, že se má zachovat pouze posledních 10 záloh a zbytek smazat. K tomu slouží místo časového údaje direktiva B (backups).

```
$ rdiff-backup --remove-older-than 10B --force /mnt/backup/adresar
```

K prohledávání již stávajících záloh slouží parametr -l.

```
$ rdiff-backup -l /mnt/backup/adresar
```

Lze si také vyfiltrovat poslední změny na záloze, například za poslední měsíc pomocí parametru --list-changed-since 1M a pomocí --compare lze porovnávat změny mezi zálohou a adresářem. Pokud nějaké změny jsou, bude na místě provést zálohu.

```
$ rdiff-backup --compare /home/adresar /mnt/backup/adresar
```

Pro kompletní popis průběhu zálohy přes plánovač cron bude sloužit následující kód. Zde je třeba si nejprve vytvořit na zálohovacím serveru (backupserver.company.com) uživatele a skupinu, kterým budeme zálohy provádět. Tento uživatel nesmí mít přístup k příkazovému interpretu. Záloha uživatelem root není z bezpečnostních důvodů doporučena.

```
$ groupadd -g 5000 backup
$ useradd -u 3500 -s /bin/false -d /backup -m -c "zalohovaci-uzivatel" -g backup backup
```

Po přihlášení jako uživatel backup

```
$ sudo -m backup
```

je třeba vygenerovat ssh-klíč a vytvořit konfigurační nastavení pro bezpečný přenos souborů. Klíče je třeba uložit do složky /backup/.ssh/id_rsa na zálohovacím serveru. Kvůli automatizaci není potřeba zadávat passphrase, jinak by se muselo zadávat heslo při každém započetí zálohy.

```
$ cd /backup
$ ssh-keygen
```

Konfigurační soubor bude uložen v /backup/config a bude vypadat následovně.

```
host fileserver
hostname sambaserver.company.com
user root
identityfile /backup/.ssh/id_rsa
compression yes
cipher blowfish
protocol 2
```

Nakonec je třeba nastavit patřičná práva na soubory.

```
$ chmod -R 700 /backup/.ssh/
```

Ze zálohovacího serveru je třeba zkopírovat veřejný klíč na souborový server. Veřejný klíč se na souborovém serveru uloží do souboru /root/.ssh/authorized_keys.

```
$ ssh-copy-id -i ~/.ssh/id_rsa.pub root@sambaserver.example.com
```

Nyní je třeba se přihlásit jako root na sambaserver.company.com a otevřít soubor /root/.ssh/authorized_keys, kde přidáme k řádku

```
$ ssh-rsa AAAAB3Nza[...]W1go9M= backup@backup
```

následující kód tak, aby bylo vše v jednom řádku oddělené mezerou, Výsledný řádek bude vypadat takto

```
command="rdiff-backup --server --restrict-read-only /",from="backupserver.company.com",no-port-forwarding,no-X11-forwarding,no-pty
 ssh-rda AAAAB3Nza[...]W1go9M = backup@backup
```

Příkaz uvedený níže, se provede vždy, když se přihlásí přes SSH uživatel backup z backupserver.company.com na sambaserver.company.com. Direktiva --restrict-read-only / zajišťuje pouze čtecí práva pro uživatele backup na serveru sambaserver.company.com.

```
$ rdiff-backup --server --restrict-read-only /
```

Posledními kroky k dokončení automatické zálohy je nastavení práv na /root/ssh

```
$ chmod -R 700 /root/ssh
```

Soubor /etc/ssh/sshd_config by měl obsahovat následující řádky

```
RSAAuthentication yes
PubkeyAuthentication yes
```

Pokud tyto řádky neobsahuje, je třeba je takto změnit a po změně provést restart SSH.

```
$ /etc/init.d/ssh restart
```

Nyní již záloha probíhá tak jak má a je třeba jí jen automatizovat. V tabulce úloh cronu je třeba přidat novou položku, která bude provádět zálohu celého souborového serveru, s výjimkou několika položek. Záloha se bude provádět v jednu hodinu a třicet minut ráno každý den. [9,10]

```
$ crontab -e
30 1 * * * /usr/bin/rdiff-backup --exclude /tmp --exclude /mnt --exclude /proc --exclude
/dev --exclude /cdrom --exclude /floppy fileserver::/ /backup/fileserver
```

Celý příkaz musí být na jednom řádku.

Konfigurace firewallu

Správná konfigurace firewallu je jednou z nejdůležitějších částí bezpečnostních opatření. Jelikož firewall představuje vstupní bránu do a z firemní sítě, je třeba pečlivě nastavit pravidla propouštění paketů. V roli firewallu bude jednoduchý počítač s Linuxem používající nástroj IPtables. Skript pro pravidla IPtables bude spouštěn při startu serveru. Politika firewallu by měla být taková, že pouze minimum nezbytně nutných služeb bude povoleno a podle potřeby se přidávají další.

Jako základní pomůcku pro určení IP adresy, masky a všesměrové sítě poslouží jednoduchý regulární výraz. Filtrované výrazy se mohou lišit v důsledku různých verzí distribuce, proto je dobré nejprve zavolat příkaz ifconfig a výrazy případně upravit.

```
ifconfig eth0 | grep ‘adr:‘ | sed ‘s/.*adr:\([^ ]\).*/\1/‘
ifconfig eth0 | grep ‘směr:‘ | sed ‘s/.*směr:\([^ ]\).*/\1/‘
ifconfig eth0 | grep ‘Maska:‘ | sed ‘s/.*Maska:\([^ ]\)/\1/‘
```

Firewall jakožto vstupní a řídící zařízení má dvě rozhraní, vnitřní a vnější. Třetím rozhraním je demilitarizovaná zóna. S ním spojené jsou vnitřní a vnější IP adresy, všesměrové adresy, masky a adresy sítě, které se definují na začátku skriptu. Velkou chybou při zabezpečování sítí je spouštění síťových rozhraní ještě před zavedením bezpečnostních pravidel. Linux v základním nastavení propouští všechny pakety, čímž vzniká bezpečnostní díra, byť jen na malý moment, kterou může útočník zneužít. Firewallový skript je proto třeba spouštět automaticky při spouštění systému.

```
#!/bin/sh
#
# Externí rozhraní
EXTIF=eth0
# Interní rozhraní
INTIF=eth1
# Rozhraní DMZ
DMZIF=eth2
# DMZ servery
# doplnit IP podle potřeby, pokud se nechá prázdné, nebude brán v potaz
# napr: HTTP_IP="192.168.5.1"
HTTP_IP=""
HTTPS_IP=""
MAIL_IP=""
DNS1_IP=""
POP3_IP=""
POP3S_IP=""
IMAP_IP=""
IMAPS_IP=""
ISPDNS1=""
ISPMAIL1=""
#Externí nastaveni
EXTIP=externi_IP
EXTBC=externi_broadcast_adresa
EXTM=externi_maska
EXTSIT=EXTIP/EXTM
# Interní nastaveni
INTIP=interni_IP
INTBC=interni_broadcast_adresa
INTM=interni_maska
INTSIT=INTIP/INTM
# DMZ nastaveni
DMZIP=DMZ_IP
DMZBC=DMZ_broadcast_adresa
DMZM=DMZ_maska
DMZSIT=DMZIP/DMZM

#Prvním krokem před samotnou konfigurací iptables je upravit jádro, kde zakážeme
#určité služby, které umožňují DoS útoky.
echo 1 > /proc/sys/net/ipv4/tcp_syncookies
```

```
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcast
#Ochranu proti falšování zdrojové adresy paketů, lze zapnout následujícím skriptem,
#který modifikuje jádro.
for i in /proc/sys/net/ipv4/conf/*/rpfilter;
do
  echo 1 > $i
done
#Následující skript zakáže útočníkovi směrování zdrojem, což by mu umožnilo vést útoky
#ze sítě na jiné stroje.
for i in /proc/sys/net/ipv4/conf/*/accept_source_route;
do
  echo 0 > $i
done

for i in /proc/sys/net/ipv4/conf/*/accept_redirects;
do
  echo 0 > $i
done

echo 1 > /proc/sys/net/ipv4/ip_forward

#zapnout defragmentaci paketů v jádře - obrana proti útoku fragmentací
echo 1 > /proc/sys/net/ipv4/ip_always_defrag

#Vytváření samotných iptables bude prováděno pomocí parametru -A (append), které
#umožní přidávání pravidel na první místo v seznamu.
# Zahození všech paketů a vyčištění pravidel
# pravidla DROP musí být zavedena, protože po vyčištění seznamu vzniká časové okno,
# než se nastaví potřebná pravidla
iptables    -P    OUTPUT     DROP
iptables    -P    INPUT      DROP
iptables    -P    FORWARD    DROP
iptables    -F
#Na začátku listu budou pravidla, která zahodí a zaevidují všechny
#pakety. Postupně budou přidávána pravidla pro povolování služeb.
iptables    -N    DROPL      2 > /dev/null
iptables    -A    DROPL      -j LOG --log-prefix “DROPL:“
iptables    -A    DROPL      -j DROP
iptables    -N    REJECTL    2 > /dev/null
iptables    -A    REJECTL    -j LOG --log-prefix “REJECTL:“
iptables    -A    REJECTL    -j REJECT
#Povolení loopback pro všechna rozhrani
iptables    -A    INPUT  -i    lo      -s    127.0.0.1     -j    ACCEPT
iptables    -A    INPUT  -i    lo      -s    $EXTIP        -j    ACCEPT
iptables    -A    INPUT  -i    lo      -s    $INTIP        -j    ACCEPT
#Blokování všesměrového vysílání
iptables    -A    INPUT  -i    $EXTIF  -d    $EXTBC        -j    DROPL
iptables    -A    INPUT  -i    $INTIF  -d    $INTBC        -j    DROPL
iptables    -A    INPUT  -i    $DMZIF  -d    $DMZBC        -j    DROPL
iptables    -A    OUTPUT -o    $EXTIF  -d    $EXTBC        -j    DROPL
iptables    -A    OUTPUT -o    $INTIF  -d    $INTBC        -j    DROPL
iptables    -A    OUTPUT -o    $DMZIF  -d    $DMZBC        -j    DROPL
```

```
iptables -A FORWARD -o $EXTIF -d $EXTBC -j DROPL
iptables -A FORWARD -o $INTIF -d $INTBC -j DROPL
iptables -A FORWARD -o $DMZIF -d $DMZBC -j DROPL
#Zablokování všech paketů pocházejících z vnější sítě, které nemají jako cílovou adresu
#$EXTIP
iptables -A INPUT -i $EXTIF -d ! $EXTIP -j DROPL
#Blokování komunikace v interní síti se špatnou síťovou adresou
iptables -A INPUT -i $INTIF -s ! $INTSIT -j DROPL
iptables -A OUTPUT -o $INTIF -d ! $INTSIT -j DROPL
iptables -A FORWARD -i $INTIF -s ! $INTSIT -j DROPL
iptables -A FORWARD -o $INTIF -o ! $INTSIT -j DROPL
#Pro jistotu kontrolní pravidlo
iptables -A OUTPUT -o $EXTIF -s ! $EXTSIT -j DROPL
# Blokování DMZ se špatnou síťovou adresou
iptables -A INPUT -i $DMZIF -s ! $DMZSIT -j DROPL
iptables -A OUTPUT -o $DMZIF -d ! $DMZSIT -j DROPL
iptables -A FORWARD -i $DMZIF -s ! $DMZSIT -j DROPL
iptables -A FORWARD -o $DMZIF -d ! $DMZSIT -j DROPL
#Blokování odchozího ICMP (kromě ping)
iptables -A OUTPUT -o $EXTIF -p icmp \ --icmp-type ! 8 -j DROPL
iptables -A FORWARD -o $EXTIF -p icmp \ --icmp-type ! 8 -j DROPL

#pripojeni souboru fw.trouble který je generován programem PortSentry
. /etc/rc.d/fw.trouble

#Blokování služeb a portů, které jsou častým cílem útočníků
#
# BOTH - výčet služeb pro TCP a UDP (společně)
# 0 - tcpmux
#13 - daytime
#98 - linuxconf
#111 - sunrpc
#137:139,445 - Microsoft
#161:162 - SNMP
#3128,8000,8008,8080 - Squid
#1214 - Morpheus, KaZaA
#2049 - NFS
#3049,12345,65535 - Trojské koně
#1999,4329,6346 - Běžné útoky
#editace portů podle potřeby se provádí zde
BOTHBLOK="0:1 13 98 11 137:139 161:162 445 1214 199 2049 3049 4329 6346 \
 3128 8000 80 8 8080 12345 65535"

#Porty pro TCP:
#98 - Linuxconf
#512:515 - rexec, rlogin, rsh, printer(lpd)
#1080 - proxy server Socks
#6000 - X Window
#6112 - Sun/HP CDE
#editace portů podle potřeby se provádí zde
TCPBLOK="$BOTHBLOK 98 512:515 1080 6000:6009 6012"
#Porty pro UDP:
```

```
#520 - RIP
#9000 - nebezpečný port
#517:518 - talk, kalk
#editace portů podle potřeby se provádí zde
UDPBLOK="$BOTHBLOK 520 123 517:518 1427 9000"
echo -n "FW: Blokovani TCP portů "
for i in $TCPBLOK;
do
        echo -n "$i "
        iptables        -A      INPUT  -p       tcp     --dport   $i    -j       DROPL
        iptables        -A      OUTPUT -p       tcp     --dport   $i    -j       DROPL
        iptables        -A      FORWARD -p      tcp     --dport   $i    -j       DROPL
done
echo " "

echo -n "FW: Blokovani UDP portů "
for i in $UDPBLOK;
do
        echo -n "$i "
        iptables        -A      INPUT  -p       udp     --dport   $i    -j       DROPL
        iptables        -A      OUTPUT -p       udp     --dport   $i    -j       DROPL
        iptables        -A      FORWARD -p      udp     --dport   $i    -j       DROPL
done
echo " "

#Povolení následujících služeb
#editace služeb podle potřeby se provádí zde
#11371 - PGP/GPG
TCPSLUZBY="domain ssh http http mail pop3 pop3s imap3 imaps 11371 vpn"
UDPSLUZBY="domain"

echo -n "FW: Povolení služeb TCP: "
for i in $TCPSLUZBY;
do
        echo -n "$i "
        iptables        -A      OUTPUT          -o      $EXTIF -p tcp   -s      $EXTIP \
                --dport   $i    --syn   -m      state   --state NEW    -j      ACCEPT
        iptables        -A      FORWARD         -i      $INTIF -p tcp   -s      $INTSIT \
                --dport   $i    --syn   -m      state   --state NEW    -j      ACCEPT
done
echo " "

echo -n "FW: Povolení služeb UDP: "
for i in $UDPSLUZBY;
do
        echo -n "$i "
        iptables        -A      OUTPUT          -o      $EXTIF -p udp   -s      $EXTIP \
                --dport   $i    -m      state   --state NEW    -j      ACCEPT
        iptables        -A      FORWARD         -i      $INTIF -p udp   -s      $INTSIT \
                --dport   $i    -m      state   --state NEW    -j      ACCEPT
done
echo " "
```

```
#povolení portů pro Sambu 137:139,445
iptables        -A      INPUT  -p tcp  -s $INTSIT      --dport 137:139  -j      ACCEPT
iptables        -A      INPUT  -p tcp                  --dport 137:139  -j      LOG
iptables        -A      INPUT  -p tcp                  --dport 137:139  -j      DROP
iptables        -A      INPUT  -p tcp  -d $INTSIT      --sport 137:139  -j      ACCEPT
iptables        -A      INPUT  -p tcp                  --sport 137:139  -j      LOG
iptables        -A      INPUT  -p tcp                  --sport 137:139  -j      DROP

iptables        -A      INPUT  -p udp  -s $INTSIT      --dport 137:139  -j      ACCEPT
iptables        -A      INPUT  -p udp                  --dport 137:139  -j      LOG
iptables        -A      INPUT  -p udp                  --dport 137:139  -j      DROP
iptables        -A      INPUT  -p udp  -d $INTSIT      --sport 137:139  -j      ACCEPT
iptables        -A      INPUT  -p udp                  --sport 137:139  -j      LOG
iptables        -A      INPUT  -p udp                  --sport 137:139  -j      DROP

iptables        -A      INPUT  -p tcp  -s $INTSIT      --dport 445  -j   ACCEPT
iptables        -A      INPUT  -p tcp                  --dport 445  -j   LOG
iptables        -A      INPUT  -p tcp                  --dport 445  -j   DROP
iptables        -A      INPUT  -p tcp  -d $INTSIT      --sport 445  -j   ACCEPT
iptables        -A      INPUT  -p tcp                  --sport 445  -j   LOG
iptables        -A      INPUT  -p tcp                  --sport 445  -j   DROP

iptables        -A      INPUT  -p udp  -s $INTSIT      --dport 445  -j   ACCEPT
iptables        -A      INPUT  -p udp                  --dport 445  -j   LOG
iptables        -A      INPUT  -p udp                  --dport 445  -j   DROP
iptables        -A      INPUT  -p udp  -d $INTSIT      --sport 445  -j   ACCEPT
iptables        -A      INPUT  -p udp                  --sport 445  -j   LOG
iptables        -A      INPUT  -p udp                  --sport 445  -j   DROP

#povolení ping směrem ven
iptables        -A      OUTPUT         -o      $EXTIF -p icmp -s       $EXTIP \
                --icmp-type     8       -m      state   --state NEW   -j       ACCEPT
iptables        -A      FORWARD        -i      $INTIF -p icmp -s       $INTSIT \
                --icmp-type     8       -m      state   --state NEW   -j       ACCEPT
#povolení ping pro interní systémy
iptables        -A      OUTPUT         -o      $INTIF -p icmp -s       $INTIP \
                --icmp-type     8       -m      state   --state NEW   -j       ACCEPT

#vyvržení služby auth, pro rychlou odezvu poštovnímu serveru
iptables        -A      INPUT  -p       tcp     --dport         auth    -j       REJECT

# Přesměrování auth požadavků do DMZ
# Může se omezit pouze na server, které posílají emaily do Internetu
iptables -t nat -A PREROUTING -p tcp --dport auth --syn -m state \
  --state NEW -d $DMZSIT -j DNAT --to-destination $EXTIP

# Povolit zadané DMZ servery
if [ "$DNS1_IP" != "" ] ; then
  iptables -A FORWARD -i ! $DMZIF -p udp -d $DNS1_IP \
    --dport domain      -m state --state NEW -j ACCEPT
fi
```

```
# Obvykle se nepovoluje TCP DNS  z důvodu předcházení „Zone Transfers“
if [ "$HTTP_IP" != "" ] ; then
  iptables -A FORWARD -i ! $DMZIF -p tcp -d $HTTP_IP \
    --dport http   --syn -m state --state NEW -j ACCEPT
fi
if [ "$HTTPS_IP" != "" ] ; then
  iptables -A FORWARD -i ! $DMZIF -p tcp -d $HTTPS_IP \
    --dport https  --syn -m state --state NEW -j ACCEPT
fi
if [ "$MAIL_IP" != "" ] ; then
  iptables -A FORWARD -i ! $DMZIF -p tcp -d $MAIL_IP \
    --dport smtp   --syn -m state --state NEW -j ACCEPT
fi
if [ "$POP3_IP" != "" ] ; then
  iptables -A FORWARD -i ! $DMZIF -p tcp -d $POP3_IP \
    --dport pop3   --syn -m state --state NEW -j ACCEPT
fi
if [ "$POP3S_IP" != "" ] ; then
  iptables -A FORWARD -i ! $DMZIF -p tcp -d $POP3S_IP \
    --dport pop3s  --syn -m state --state NEW -j ACCEPT
fi
if [ "$IMAP3_IP" != "" ] ; then
  iptables -A FORWARD -i ! $DMZIF -p tcp -d $IMAP3_IP \
    --dport imap3  --syn -m state --state NEW -j ACCEPT
fi
if [ "$IMAP3S_IP" != "" ] ; then
  iptables -A FORWARD -i ! $DMZIF -p tcp -d $IMAP3S_IP \
    --dport imap3s --syn -m state --state NEW -j ACCEPT
fi

# Povolí DMZ systémům používat následující služby
if [ "$ISPDNS1" != "" ] ; then
  iptables -A FORWARD -i $DMZIF -p udp -s $DMZNET \
    --dport domain -d $ISPDNS1 -m state --state NEW -j ACCEPT
fi
if [ "$ISPMAIL1" != "" ] ; then
  iptables -A FORWARD -i $DMZIF -p tcp -s $DMZNET \
    --dport smtp   -d $ISPMAIL1 -m state --state NEW -j ACCEPT
fi

iptables        -t nat   -A       PREROUTING                  -j ACCEPT
iptables        -t nat   -A       POSTROUTING                 -j ACCEPT
iptables        -t nat   -A       OUTPUT                      -j ACCEPT

#povolení ssh + logování přístupu
#nahradit <IPADDR> za vlastní IP adresu firewallu
iptables        -A       INPUT  -p       tcp      -m      tcp      --dport          22      \
        -m state --state INVALID,NEW -j LOG --log-prefix "SSH connection: "
iptables        -A       INPUT -s  <IPADDR> -p tcp -m tcp      --dport          22      \
        -m state --state NEW,ESTABLISHED -j ACCEPT
```

```
#Povolení zbytku paketů - odpovědi, poslední paket z trojcestného navazování
#komunikace, …
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT

#Zahodit a zalogovat ostatní
iptables -A INPUT -j DROPL
iptables -A OUTPUT -j REJECTL
iptables -A FORWARD -j DROPL

# Povolit Proxy ARP pro pakety z externího rozhraní
echo "Povolovani Proxy ARP z externiho rozhrani"
echo 1 > /proc/sys/net/ipv4/conf/$EXTIF/proxy_arp
```

Z tohoto skriptu je třeba udělat symbolický odkaz do adresáře, ze kterého bude spouštěn při startu systému. [1]

Konfigurace bezpečnostních služeb

Tyto služby napomáhají udržovat bezpečnost sítě a zjišťovat případné nedostatky. Výčet níže uvedených služeb není definitivní, lze je dále kombinovat s jinými programy, nahradit nebo úplně vynechat. Programy jsou vybírány tak, aby splňovaly podmínky otevřeného zdrojového kódu a byly zdarma.

Konfigurace a použití ArpWatch

ArpWatch je program, sloužící k monitorování ARP provozu na počítačové síti. Jeho předností je, že umí uchovávat dvojice IP a MAC adres do databáze spolu s časovým údajem kdy tuto dvojici objevil. Je to vynikající program, k zjišťování útoku ARPspoofing.

Instalace programu probíhá klasickou cestou přes apt-get.

```
$ sudo apt-get install arpwatch
```

Soubory k ukládání dvojic adres je třeba vytvořit do adresáře /var/lib/arpwatch/. Pro každé rozhraní je třeba vytvořit zvláštní soubor a nastavit mu práva 600 a vlastníka roota.

```
$ cd /var/log/arpwatch/
$ touch eth0, eth1, eth2
$ chmod 600 eth0 eth1 eth2
```

Konfigurační soubor je v adresáři /etc/arpwatch.conf ale ten není třeba konfigurovat. Stačí pouze správně spouštět instance programu.

Pro každé rozhraní nastavíme jeho logovací soubor a rozhraní do skriptu, přes který se bude pouštět arpwatch.

```
#!/bin/sh
echo "Spouštění ArpWatch ... "
AWATCH=/usr/sbin/arpwatch
ARPARG="-0 -m admin@company.com"
$AWATCH -f /var/log/eth0 -i eth0 -w "arpwatch(eth0)" $ARPARG
$AWATCH -f /var/log/eth1-i eth1 -w "arpwatch(eth1)" $ARPARG
$AWATCH -f /var/log/eth2 -i eth2 -w "arpwatch(eth2)" $ARPARG
```

Ověření běhu programu lze pomocí příkazu:

```
$ ps -axlww | grep arpwatch
```

Bližší informace o konfiguraci jsou k nalezení v dokumentaci programu.

Konfigurace a použití OpenVas

Program OpenVas je bezpečnostní program, provádějící audit sítě. Kontroluje známé bezpečnostní chyby, které nalezne v systému a navrhne i možnost opravy.

Instalace program se provede pomocí příkazu aptitude.

```
$ sudo aptitude -t lenny-backports install openvas-server
```

Po instalaci produktu, je třeba vytvořit certifikát.

```
$ sudo openvas-mkcert
```

K použití programu je dále nezbytně nutné vytvořit uživatele, pod kterým se bude spouštět OpenVas. Tento uživatel by neměl být v sudo nebo root skupině.

```
$ sudo openvas-adduser
```

Po zadání uživatelského jména, lze zvolit buď ověřování heslem, nebo certifikátem. Lepší volbou se jeví zvolit heslo. Po zadání hesla, program vyžádá zadat pravidla pro uživatele. Tato pravidla mají následující obecnou syntaxi:

```
deny | accept  ip/mask
default accept | deny
```

Takto lze povolovat sítě či konkrétní stanice, ze kterých lze provádět skenování. Například pro povolení skenování sítě 192.168.2.0/24 a zakázání zbytku lze použít syntaxi

```
accept          192.168.2.0/24
default         deny
```

Pro zakázání skenování stanic 192.168.2.1, 192.168.2.20 a povolení zbytku sítě lze použít syntaxi

```
deny            192.168.2.1
deny            192.168.2.20
default         accept
```

Dále je možné použít direktivu „client_ip“, která povolí skenování pouze té stanice, ze které se daný uživatel přihlašuje. Použití je následující

```
accept          client_ip
default         deny
```

Pro úplný chod a kompletní testování je třeba doinstalovat pluginy NVT.

```
$ sudo openvas-nvt-sync
```

Konfigurační soubor /etc/openvas/openvas.conf lze nastavit dle konkrétních požadavků. Standardní nastavení je ovšem dostatečné a není třeba ho upravovat. Před samotnou instalací klienta, který se musí nacházet na všech stanicích, ze kterých má být OpenVas spouštěn, je třeba inicializovat server OpenVas příkazem:

```
$ openvasd -D &
```

Instalace klienta probíhá jednoduchou formou:

```
$ sudo apt-get install openvas-client
```

Nyní lze spouštět program v grafickém rozhraní

```
$ openvas-client
```

Nebo v konzolovém rozhraní

```
$ openvas -qx <IP_cíle> 9390 username password targetfile resultfile
```

Kde username a password je uživatelské jméno a heslo, které se zadávalo při přidávání uživatele do OpenVas. Targetfile je cesta k existujícímu souboru, ze kterého se bude načítat cíl. Resultfile je soubor, do kterého se vygeneruje výsledek testu. Podrobnější návod k instalaci a konfiguraci produktu lze nalézt v jeho manuálových stránkách a oficiální dokumentaci.

Konfigurace a použití Nmap

Nástroj nmap slouží k monitorování sítě, kontroly otevřených portů a mapovaní hostitelských stanic v síti. Nmap může být použít i k testování firewallu.

Instalace nmap probíhá klasickým způsobem.

```
$ sudo apt-get install nmap
```

Pro otestování firewallových pravidel, lze spustit následující příkazy, které otestují různou propustnost paketů a otevřených portů. Je třeba pracovat uvážlivě s operátorem -T Aggressive, protože někdy způsobuje pády testovaných systémů.

```
$ nmap -P0 -sS -F -0 -T Aggressive <IP_adresa>
$ nmap -P0 -sU -F -0 -T Aggressive <IP_adresa>
$ nmap -sP -sS -0 -p 21,22,23,25,53,110,80,113,139,1024,6000 \
 -T Aggressive <INTSIT>
```

Tyto příkazy je vhodné spouštět i z jiných systémů vně a uvnitř sítě směrem na firewall. Další běžné příkazy k monitorování firemní sítě:

```
#zjišťování dostupných stanic na síti 192.168.1.0/24
$ nmap -sP -n 192.168.1.0/24
#zjišťování otevřených portů na konkrétním hostu
$ nmap -n 192.168.1.5
#zjištění verzí běžících programů
$ nmap -n -sV 192.168.1.5
#zjištění konkrétních otevřených portů na síti
$ nmap -n -p 80,443 192.168.1.0/24 | egrep "ports|open"
#zjišťování verzí programů na daném portu, například kvůli bezpečnostnímu
#opatření vůči konkrétní verzi
$ nmap -n -sV -p 139 192.168.1.0/24 | egrep "ports|139"
```

Nmap je velmi šikovný nástroj, který se bude hodit každému administrátorovi. Má velmi rozsáhlé možnosti od obyčejného zjišťování otevřených portů, přes detekci operačních systému až po podrobné zjišťování informací o dané stanici nebo síti. Podrobnější použití nmap lze nalézt v jeho manuálových stránkách. [4]

Konfigurace a použití TripWire

Nástroj TripWire vytváři otisky kontrolních součtů pro vybrané soubory a ukládá je do své databáze. Tyto otisky lze kdykoliv zkontrolovat a tak zjistit, jestli se změnily. Pokud ano, je třeba zjistit, proč byl soubor změněn.

Adresáře, ve kterých je třeba kontrolovat soubory:

```
/etc
/bin
/usr/bin
aplikace a uživatelské účty
```

Kontrola integrity by měla probíhat automaticky pomocí plánovače cron. Instalace programu se provede klasickým příkazem pomocí apt-get.

```
$ sudo apt-get install tripwire
```

Při instalaci je třeba následovat pokyny průvodce. Po přepnutí do hlavního adresáře programu TripWire, je třeba vytvořit bezpečnou verzi konfiguračního souboru a vygenerovat databázi.

```
$ cd /usr/sbin
$ sudo ./twadmin --create-polfile /etc/tripwire/twpol.txt
$ sudo ./twadmin --init
```

Po vygenerování databáze je TripWire již funkční a zaznamenává veškeré změny na souborech, které jsou pod dozorem. Nyní je třeba pouze otestovat posílání emailu a nastavit automatickou kontrolu do plánovače cron.

Otestování emailu:

```
$ ./tripwire --test --email root
```

Spouštění programu:

```
$ ./tripwire -m c
```

nebo

```
$ ./tripwire --check
```

Spuštění v plánovači cron. Akce bude spouštěna každý den v šest hodin ráno a výsledky se odešlou uživateli root na email (celý příkaz musí být na jednom řádku).

```
0 6 * * * /usr/sbin/tripwire -m c -r /usr/sbin/report/nocni-kontrola.twr
2>&1 | mail -s TRIPWIRE root
```

Konfigurace a použití PortSentry

PortSentry zastupuje sekci IDS ve firemní síti. IDS je zkratkou pro Intrusion Detection System, čili systém detekce narušení. Jde o program, který téměř v reálném čase reaguje na podněty zvenčí předem definovaným chováním.

Jedná se především o monitorování přednastavených portů, na které útočníci běžně útočí. Pokud PortSentry zjistí, že se někdo dobývá na daný port, provede zablokování přístupu pro jeho IP adresu.

Stažení programu se provede příkazem

```
$ wget
http://downloads.sourceforge.net/project/sentrytools/portsentry%201.x/portsentry-1.2/portsentry-1.2.tar.gz
```

Po rozbalení archivu je třeba se přesunout do složky s konfiguračním souborem portsentry.conf. Zde je třeba přenastavit následující řádky.

```
ADVANCED_PORTS_TCP="6010"
ADVANCED_PORTS_UDP="1024"
ADVANCED_EXCLUDE_TCP="113,139,37"
ADVANCED_EXCLUDE_UDP="520,138,137,67,123,37"
BLOCK_UDP="2“
BLOCK_TCP="2"

KILL_RUN_CMD="echo /sbin/iptables -I INPUT 1 -s $TARGET$ -j \
DROP >> /cesta/k/soubor/fw.trouble"

PORT_BANNER="** UNAUTHORIZED ACCESS PROHIBITED *** \
YOUR CONNECTION ATTEMPT HAS BEEN LOGGED. GO AWAY."
```

BLOCK_UDP a TCP se nastavila hodnota 2, aby se mohl volat externí příkaz v případě zjištění detekce. Jako externí příkaz je volán příkaz pro ip tables, který zablokuje danou IP adresu a pravidlo zapíše do souboru fw.trouble, který se načítá ve skriptu pro firewall. Do souboru fw.trouble se ukládá z důvodu neobnovování přístupu útočníka po restartu serveru. Poslední změnou je PORT_BANNER, který definuje oznámení útočníkovi. Položka může být libovolná.

Pro bezpečné nastavení je třeba nastavit mod 600 na konfigurační soubor.

```
$ chmod 600 portsentry.conf
```

Do souboru portsentry.ignore lze nastavit hosty nebo celé sítě, které bude program PortSentry ignorovat. Jedná se o spřátelené systémy, kterým chceme umožnit přístup, bez omezení a logování. Zde se musí vyskytovat položka 127.0.0.1 a 0.0.0.0. Při přidávání sítí nebo hostitelských počítačů není doporučeno vkládat celou firemní síť.

Opět je třeba zabezpečit soubor portsentry.ignore nastavením modu 600.

```
$ chmod 600 portsentry.ignore
```

Sestavení programu probíhá příkazem

```
$ make linux
```

a instalace příkazem

```
$ make install
```

PortSentry je vhodné spouštět automaticky hned po startu systému a to ve dvou režimech

```
$ portsentry -atcp //pokrocile neviditelne sledovani TCP
$ portsentry -audp //pokrocile neviditelne sledovani UDP
```

Kontrolu logování lze sledovat v souboru /var/log/messages příkazem

```
$ tail -20f /var/log/messages
```

Pro otestování správné funkčnosti se může administrátor zkusit připojit na firewall příkazem

```
$ telnet firewall 6000
```

A zkontrolovat, jestli program PortSentry zareagoval správně a přidal pravidlo do iptables.

```
$ iptables -L -n -v
```